FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

Demand forecasting in a multi-specialty hospital setting: a comparative study of machine learning and classical statistical methods

Carlos Miguel Ferreira Alves

Mestrado Integrado em Engenharia Informática e Computação

Supervisor: Prof. Mário Amorim Lopes

July 27, 2018

Demand forecasting in a multi-specialty hospital setting: a comparative study of machine learning and classical statistical methods

Carlos Miguel Ferreira Alves

Mestrado Integrado em Engenharia Informática e Computação

July 27, 2018

Abstract

The inventory management is a key component of the hospital logistics, making sure that medicines as well as all medical supplies get to the patient within the appropriate time. Currently, hospitals do this management in a rudimentary manner, not always making good use of the best practices, and without resorting to mathematical or statistical models that help anticipate seasonal changes and needs. It's possible to obtain more reliable predictions using machine learning techniques and better results from more systematic statistical methods.

The *Knowlogis* project consists of the development of a dashboard that will provide a valuable hospital management tool, providing support in the decision making processes of the hospital management team. This system will incorporate a layer which will carry out the forecasting of material requirements, thus anticipating problems which might not have been foreseen by the staff. Carrying out an accurate inventory forecast is useful for inventory management policies become more efficient, significantly minimizing risks not only of excess stock and the associated costs, but also reducing stock shortages.

Using data on the consumption history of a large urban multi-specialty Portuguese hospital, categorized by demand patterns, we study the applicability of machine learning algorithms to forecasting the demand for hospital consumables and compare them to more traditional methods. In particular, we explore the performance of methods such as Recurrent Neural Networks (RNN), Support Vector Machines (SVM) and Random Forest (RF) with regards to parametric methods such as the Holt-Winters Exponential Smoothing or Auto-regressive Integrated Moving Average (ARIMA). The purpose of this dissertation is to evaluate the performance of models using accuracy metrics and computational complexity. After comparing, we found that machine learning algorithms are the ones that yield the smallest root mean squared errors. It was also possible to verify that the items' category directly influences the performance of the forecasting models.

Resumo

A gestão de inventário é uma componente fulcral da logística hospitalar, assegurando que os medicamentos e o material clínico chegam ao paciente na altura devida. Atualmente, os sistemas hospitalares fazem esta gestão de uma forma rudimentar, nem sempre fazendo bom uso das melhores práticas, e sem recorrer a modelos que ajudem a antecipar necessidades sazonais. É possível obter previsões mais fiáveis através de técnicas de *machine learning* e de métodos estatísticos mais sistemáticos.

O projeto *Knowlogis* consiste no desenvolvimento de um *dashboard* de apoio à decisão num contexto de gestão hospitalar. Este sistema incorporará uma camada responsável pela previsão de necessidades, antecipando problemas que não são percetíveis por um humano. Realizar uma previsão precisa de inventário visa tornar os modelos de gestão de inventário mais eficazes, minimizando significativamente problemas de excesso de stock e os seus custos associados, mas também reduzindo as taxas de rutura.

Usando dados relativos ao histórico de consumos de um grande hospital urbano português, categorizados por padrões de consumo, foi estudada a aplicabilidade dos algoritmos de *machine learning* para realizar previsões de necessidades de material hospitalar e compará-los com métodos estatísticos. Foram analisadas as performances de algoritmos como *Recurrent Neural Networks* (RNN), *Support Vector Machines* (SVM) e *Random Forest* (RF) e comparadas com métodos paramétricos como *Holt-Winters Exponential Smoothing* ou *Auto-regressive Integrated Moving Average* (ARIMA). O objectivo desta dissertação é avaliar a performance dos modelos usando métricas de precisão e a complexidade computacional. Depois de realizada a comparação, concluiu-se que os algoritmos de *machine learning* foram aqueles que obtiveram menores erros quadráticos médios. Constatou-se ainda que as categorias dos itens analisados influencia diretamente a performance dos algoritmos de previsão.

Acknowledgements

Firstly I would like to thank to my mother, which is the most important person of my life, for giving me all the love, the support and advices in the hardest moments of my life. I also would like to thank my whole family.

Secondly, I have to thank to my friends. Without them it would not be possible. They made this life chapter the most beautiful one. All the good moments we spent together, the support. Thank you all guys.

Thirdly, I would like to thank to my girlfriend who is my partner for all the love and support.

Lastly, a special thanks to my supervisor who helped me a lot and advised me when was necessary.

Carlos Alves

“Good, better, best. Never let it rest.
’Til your good is better and your better is best.”

St. Jerome

Contents

List of Figures

LIST OF FIGURES

List of Tables

LIST OF TABLES

Abbreviations

ARIMA	Autoregressive Integrated Moving Average
BN	Bayesian Networks
KNN	k-Nearest Neighbour
LR	Linear Regression
MA	Moving Average
MAAPE	Mean Arctangent Absolute Percentage Error
MAE	Mean Absolute Error
MAPE	Mean Absolute Percentage Error
MLR	Multiple Linear Regression
NRMSE	Normalized Root Mean Squared Error
RF	Random Forest
RMSE	Root Mean Squared Error
RNN	Recurrent Neural Network
SVM	Support Vector Machines

Chapter 1

Introduction

1.1 Context

Inventory management is a key component of hospital logistics, ensuring that medicines and medical consumables reach the patient timeously.

The hospital industry is under constant pressure due to the possibility of stock shortages of medical supplies or other consumer products, when these are required, which in turn can result in patient dissatisfaction [Tang et al., 2016]. According to Tang et al. [2016], in order to be able to provide a service with some degree of quality, it is necessary that the personnel responsible for stock replenishment have the required skill set as well as the required support tools to maintain the optimized stock levels required to meet the general requirements. Most hospital systems currently in use in this field, do so in a rudimentary may, not always following to the best practices.

With regards to requirements forecasts, most of the methods used, if and when they exist, are fairly rudimentary, and are mainly based on historical consumption data or even the intuitive verbal advice provided by experienced personnel in this area [Kwon et al., 2016].

The use of management tools can aid in the decision making processes, and the use of these forecast models can lead to the improvement of customer service standards, inventory management and planning, but above all, be a valuable aid in selecting the best management strategy, if correctly implemented Silver et al. [1998].

Methods that rely only on historical data manipulation and do not use reliable forecast models have a very low capacity for producing accurate and useful prediction results, which renders the system ineffective. Important factors that assist in this forecasting process are factors that relate to seasonality and consumption trends that are not normally taken into account. The use of more advanced analysis methods can contribute to the improvement of the forecasting results.

1.2 Motivation and goals

The purpose of this research is to attempt to identify which forecasting algorithms will provide the best results when applied in a hospital context. By implementing a reliable forecasting system of requirements, we will be able to make the inventory management models more efficient. This improvement creates a great positive impact on the health care system, as it contributes to the reduction of stock shortages, minimizes the problems associated with excessive stock, reducing the overall stock holding costs, shorten waiting times and improve overall client satisfaction. The optimized combination of these factors will contribute to a greater improvement in the quality of health systems in general.

1.3 *Knowlogis*

The *Knowlogis* project consists of the development of a dashboard that will provide support to hospital management and other decision makers. This system will incorporate an intelligent data layer which will attempt to solve the problem of inventory forecasting, inventory management and general hospital management. This layer is considered intelligent as it is constantly analyzing data inputs, which are representative of the quality of the hospital service and has the ability to predict future needs, thus anticipating problems which would not normally have been foreseen by a human. *Knowlogis* helps in the general decision making processes of hospital management as it provides preventative as well as suggests corrective actions.

Its objective is to transform a logistics management based on deterministic models to a management based on a predictive, efficient and intelligent model. Figure 1.1 represents the conceptual model of *Knowlogis*.

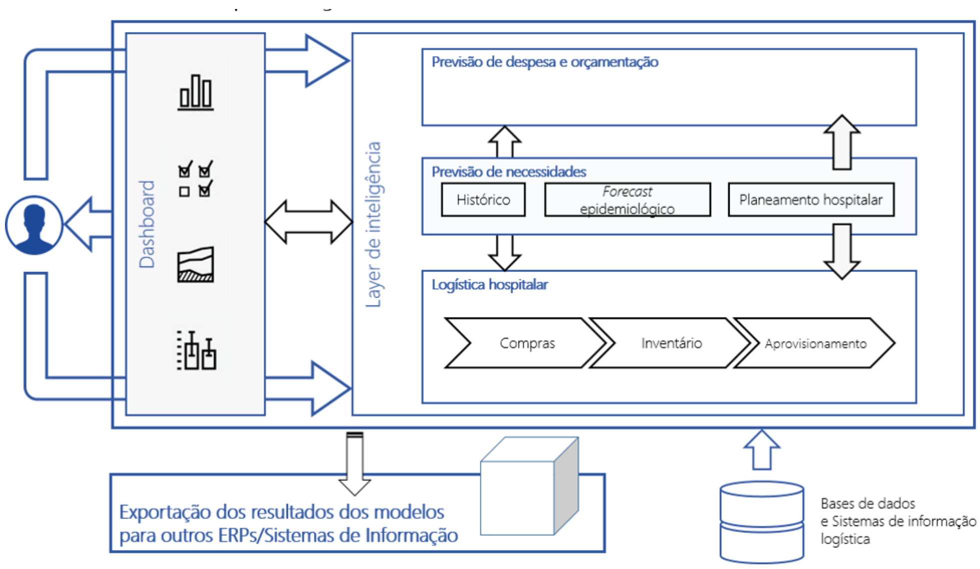


Figure 1.1: Conceptual model of *Knowlogis*

1.4 Document Structure

This report includes four main chapters and the content of each one is explained bellow.

Chapter 2 describes the existing literature regarding time series forecast techniques and algorithms. For a better understanding of the approach and implementation that were used, several concepts are explained.

Chapter 3 describes all the details concerning the implementation phase, including data preparation and split, model calibration and performance evaluation.

In chapter 4 it is included the experimental results for each algorithm and a discussion that contains the main conclusions.

Finally, chapter 5 contains the conclusions of this project and the future work with details about what could be done to improve this project.

Chapter 2

Literature Review

The analysis of consumption patterns of a specific product, for example, the consumption recorded during a day, can be considered as a time series Silver et al. [1998]. A time series is characterized by a set of data, separated by equal time intervals, and ordered sequentially in time. These series are composed of several components. They express trends, seasonality, cyclic variations, and also irregular variations (Figure 2.1). The trend refers to the numerical behaviour of the data within the specified time window of the series. The seasonality refers to similar patterns that a time series seems to display. The cyclic variations reflect recurrent behaviours, even though they don't need to be periodic. The irregular variations are characterized by short term unexplainable fluctuations. According to Assimakopoulos and Nikolopoulos [2000], the greatest difficulty is to successfully isolate the component which represents these irregular variations, and then carry out appropriate forecasts for the remaining cycle of trends.

The temporal analysis of data allows you to make future forecasts using past information. These subjects have been extensively studied over the past decades so that the developed models are now capable of making much more accurately predictions than before [Samsudin et al., 2010]. There are different approaches used to model the behaviour of a time-series. These can be divided in two main groups: approaches based on statistical methods (Figure 2.2) and approaches based on machine learning techniques (Figure 2.3). In the next chapters, we will analyze the algorithms most used in both approaches.

2.1 Statistical Methods

Statistical methods used to model the behaviour of a time-series have serious limitations, because they are not capable of producing models of complex patterns within the time window of the data [Zhou et al., 2016] . The applicability of these methods is limited , since they cannot produce satisfactory results when non-linear time series are considered [Alpaslan et al., 2012]. However, these methods are easier to understand and also to interpret the results. There are several algorithms

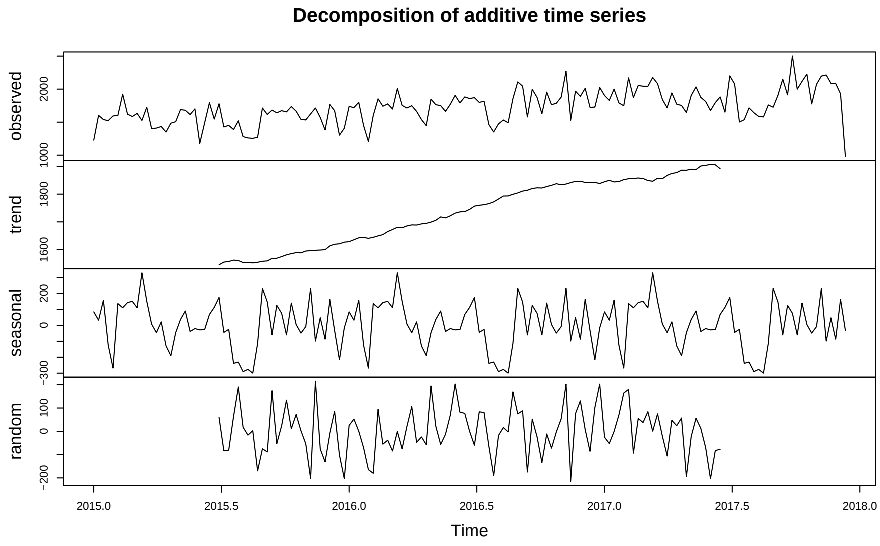


Figure 2.1: Decomposition of time series

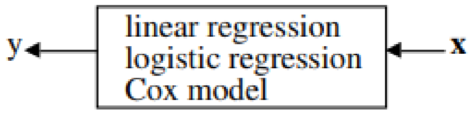


Figure 2.2: Conceptual Functioning of the Statistical Approaches [Breiman, 2001]

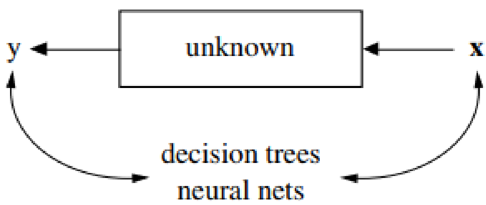


Figure 2.3: Conceptual Functioning of Machine Learning Approaches [Breiman, 2001]

developed with the objective of analyzing these time-series and creating forecasting models. Next some of these algorithms will be studied.

2.1.1 Linear Regression

Linear Regression is a frequently used algorithm, in which a model is generated that describes the relation between one quantitative independent variable and one dependent. It is defined as:

$$Y = \alpha + \beta X + \varepsilon \quad , \tag{2.1}$$

where X is the explanatory variable, Y is the dependent variable, β is the slope of the line, α is the y-intercept and ε the error term.

Multiple Linear Regression (MLR) is an extension of the Linear Regression algorithm, where the behaviour between a set of quantitative independent variables and a dependent quantitative variable are modeled. Formally, the model for multiple linear regression, given n observations, is defined as:

$$y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots \beta_p x_{ip} + \varepsilon_i \quad for\, i = 1, 2, \ldots n \tag{2.2}$$

where $x_1, x_2, \ldots, x_p$ are the explanatory variables, β_0 is the intercept, β_p is the slope and ε_i is the residual term.

Adamowski et al. [2012] used, among other algorithms, Multiple Linear Regression to predict city water consumption needs in Montreal, Canada. When the final results were evaluated, multiple linear regressions were surpassed by other algorithms, such as neural networks. The limitation identified is that MLR can not accurately predict nonlinear series. However, in the study conducted by Bon and Ng [2017], regression was the method which presented less error in predicting a drug's requirement needs.

2.1.2 Exponential Smoothing

Exponential Smoothing [Holt, 1957] is another very popular algorithm for smoothing peaks of a time series. Unlike the Simple Moving Average algorithm, which gives the same weight to past observations, when we use Exponential Smoothing we are assigning exponentially decreasing weights to observations as they become older.

By using this method of assigning weights to values, the most recent observations gain greater weight in the forecast than the older observations. There are different types of Exponential Smoothing due to the several evolutions it has undergone over the past years.

- **Single Exponential Smoothing** - this was the first method suggested to be implemented in time series that maintains a constant level, that is, that did not show a tendency or seasonality. For any time period t, the smoothed value S_t is found by computing:

$$S_t = \alpha y_{t-1} + (1-\alpha) S_{t-1} \quad 0 < \alpha \leq 1 \quad t \geq 3 \tag{2.3}$$

  where α is smoothing constant and y_{t-1} is the actual value in $t-1$ period.

- **Double Exponential Smoothing** - to address the problem of analyzing time-series with data trends, a new equation was added to deal with these cases. The use of this method is applicable when we have a time series with increasing or decreasing tendency and that does not have seasonality.

$$\begin{aligned} \text{Forecast equation} \quad & \hat{y}_{t+h} = \ell_t + hb_t \\ \text{Level equation} \quad & \ell_t = \alpha y_t + (1-\alpha)(\ell_{t-1} + b_{t-1}) \\ \text{Trend equation} \quad & b_t = \beta(\ell_t - \ell_{t-1}) + (1-\beta)b_{t-1} \quad 0 \le \beta \le 1, \end{aligned}$$

where ℓ_t denotes an estimate of the level of the series at time t, b_t denotes an estimate of the trend of the series at time t and α is the smoothing parameter for the level.

- **Triple Exponential Smoothing** - also known as Holt-Winters Method, it consists of an improvement designed to support time series with seasonality and trend. A third equation was added to deal with the seasonality of the data. Chatfield and Yar [1991] showed their results of the use of this method for time series evaluation, whose seasonality is described through multiplicative models rather than additive models. The component form for the additive method is:

$$\begin{aligned} \text{Forecast equation} \quad & \hat{y}_{t+h} = \ell_t + hb_t + s_{t+h-m(k+1)} \\ \text{Level equation} \quad & \ell_t = \alpha(y_t - s_{t-m}) + (1-\alpha)(\ell_{t-1} + b_{t-1}) \\ \text{Trend equation} \quad & b_t = \beta(\ell_t - \ell_{t-1}) + (1-\beta)b_{t-1} \\ \text{Seasonal equation} \quad & s_t = \gamma(y_t - \ell_{t-1} - b_{t-1}) + (1-\gamma)s_{t-m}, \end{aligned}$$

where m is the frequency of the seasonality and k is the integer part of $(h-1)/m$.

The component form for the multiplicative method is:

$$\begin{aligned} \text{Forecast equation} \quad & \hat{y}_{t+h|t} = (\ell_t + hb_t)s_{t+h-m(k+1)} \\ \text{Level equation} \quad & \ell_t = \alpha \frac{y_t}{s_{t-m}} + (1-\alpha)(\ell_{t-1} + b_{t-1}) \\ \text{Trend equation} \quad & b_t = \beta(\ell_t - \ell_{t-1}) + (1-\beta)b_{t-1} \\ \text{Seasonal equation} \quad & s_t = \gamma \frac{y_t}{(\ell_{t-1} + b_{t-1})} + (1-\gamma)s_{t-m} \end{aligned}$$

2.1.3 Autoregressive Integrated Moving Average

Autoregressive Integrated Moving Average (ARIMA) is an algorithm used in the prediction of non-stationary time series. It is one of the most used algorithms when modelling a time series [Zhang, 2003]. A time series is considered stationary when its properties don't depend on the time when the series was observed. Techniques of differentiation are applied to the series in order to make them stationary, thus removing trends and seasonality.

The ARIMA models then work as a kind of filter, where it tries to separate the signal from the noise, so that it is possible to make generalizations and make prediction. It integrates an autoregressive (AR) method with a moving average (MA) method. The AR(p) model is expressed as:

$$y_t = \alpha_0 + \sum_{i=1}^{p} \alpha_i y_{t-1} + \varepsilon_t \tag{2.4}$$

where α_0 is a constant, p is the lag, α_i is the coefficient of y_{t-1} and ε_t is the white noise.

The MA(q) model is defined as:

$$y_t = \alpha_0 + \sum_{i=1}^{q} b_i \varepsilon_{t-1} + \varepsilon_t \tag{2.5}$$

where α_0 is a constant, b_i is the coefficient of ε_{t-1} and ε_t is the white noise.

Hence, the ARIMA(p,d,q) model can be defined as:

$$\left(1 - \sum_{i=1}^{p} \varnothing_i L^i\right) (1-L)^d y_t = \left(1 + \sum_{i=1}^{q} \theta_i L^i\right) \varepsilon_t \tag{2.6}$$

where L is a lag operator and ε_t is the white noise.

Sato [2013] applied the ARIMA algorithm in the prediction of the development of diseases. The author presented a limitation of the model, in the treatment of epidemics, which changes the properties of the time series making it non-linear and non-stationary. A suggested solution is to reduce the time segments, using short periods to analyze the effects of each segment.

2.2 Machine Learning Methods

Classic statistical methods may not be able to come up with a solution to all the problems which can be found when trying to model a time series. Sometimes it is necessary to model complex behaviours and non-linear methods. Methods that use machine learning techniques may prove useful in dealing with such problems. The evolution of prediction algorithms was made possible thanks to the evolution of computation, which has taken place in the last few decades, that made possible the use of more complex algorithms and the advances made in algorithms like neural networks [Khalil Zadeh et al., 2014].

2.2.1 Artificial Neural Networks

A Neural Network is a computer system which simulates the learning process that takes place in the human brain. The great advantage of Neural Networks is their ability to model complex behaviours and non-linear ones in a time series. Typically, the structural design of a neural network consists of a first layer, called an input layer, a second layer, hidden layer, responsible for data processing, and a last layer, output layer, where the results are returned.

The training phase of the neural network is crucial to the success of the results. A learning algorithm is used in the network, the most popular being back-propagation. In back-propagation algorithm an update of the weights of the network connections is made in order to minimize the error. This update starts in the last layer of the network and propagates to the initial layer. With the use of this algorithm in the training phase of the neural network, it is possible to solve non-linear problems.

A single hidden layer feedforward network, which is described by a network of three layers of units connected by acyclic links, can be described as:

$$y_t = \alpha_0 + \sum_{j=1}^{q} \alpha_j g \left(\beta_{0j} + \sum_{i=1}^{p} \beta_{ij} y_{t-1} \right) + \varepsilon_t \quad , \tag{2.7}$$

where α_j and β_{ij} are the model weights, p is the number of input nodes and q is the number of hidden nodes. The logistic or sigmoid function is often used as hidden layer transfer function:

$$Sig(x) = \frac{1}{1+\exp(-x)} \tag{2.8}$$

Other activation functions can also be used such as hyperbolic tangent function:

$$Tanh(x) = \frac{\exp^x - \exp^{-x}}{\exp^x + \exp^{-x}} \tag{2.9}$$

Alpaslan et al. [2012] employed neural networks to predict time-series with the objective of trying to figure out the configuration that brings about the best results. In this study, eight time-series of real data figures were used, but from different areas. The author not only tested different activation functions but also different number of hidden layer. Other studies have been done, using neural networks for prediction, where good predictive performance was proven [Sun et al., 2008, Chang et al., 2005, Zhang et al., 2001].

2.2.2 Support Vector Machines

The algorithm Support Vector Machines (SVM) uses the projection of the data in a superior dimension to make the data linearly separable, when this separation is not possible in the original dimension of the data. For this transformation to be possible, SVM uses kernel functions. When it comes to solving classification problems, one tries to maximize the margin that separates the two classes, or to minimize the error of the margins for regression problems.

SVM can be mathematically expressed as:

$$y(x) = \mathrm{w}^T \varphi(x) + b \quad , \tag{2.10}$$

where $\varphi(x)$ represents the high dimensional feature spaces, coefficients w and b are estimated by minimizing the regularized function:

$$R(C) = \frac{1}{2}\|\mathrm{w}\|^2 + \frac{C}{n}\sum_{i=1}^{n} L_\varepsilon(d_i, y_i) \tag{2.11}$$

Regarding the kernel function, its value is equal to the inner product of two vectors X_i and X_j in the feature space $\Phi(x_i)$ and $\Phi(x_j)$. The most used kernel functions are:

$$\begin{aligned}
&\text{Linear} && \mathrm{K}(x_i, x_j) = x_i^T x_j \\
&\text{Polynomial} && \mathrm{K}(x_i, x_j) = \left(\gamma x_i^T x_j + r\right)^d \quad \gamma > 0 \\
&\text{Sigmoid} && \mathrm{K}(x_i, x_j) = \tanh(\gamma x_i^T x_j + r) \\
&\text{Radial Basis Function} && \mathrm{K}(x_i, x_j) = \exp(-\gamma \left\|x_i^T x_j\right\|^2) \quad \gamma > 0,
\end{aligned}$$

A study by Samsudin et al. [2010] analyzed the applicability of the SVM comparatively to neural networks, with multiple layers and using back-propagation as a learning algorithm in time series prediction. The results show that the SVM algorithm presented better results in the predictive task using MAE as criteria of evaluation.

2.2.3 *Ensemble Algorithms*

Ensemble Algorithms are characterized by creating several models which are then combined to give better performance. The models can be generated by applying different algorithms to the same set of training data or use the same algorithm in different sub-sets of the same training data set. After the models are generated, techniques such as voting, weighted voting and simple averaging are applied so that the results are combined.

Random Forest is an ensemble algorithm which computes multiple decisions trees by randomly sampling the training data and features for each tree, avoiding overfitting. During training phase, it uses bagging method where the general idea is the combination of learning models to increase the overall results. Predictions to a new feature vector x', can be reached by averaging the predictions from all decision trees on x'. It can be expressed as:

$$\hat{f} = \frac{1}{B}\sum_{b=1}^{B} f_b(x') \tag{2.12}$$

Random Forest has been chosen for time series forecasting by some authors. Kane et al. [2014] performed and comparative study of time series models for prediction using Random Forest and ARIMA. The objective was to use incident data to predict future occurrences of diseases events. The authors found that Random Forest outperformed the ARIMA model regarding to predictive ability. Kusiak et al. [2013] also used Random Forest to predict influent water quality. The goal was to predict the value of one influent water quality metric. The results shows that Random Forest yield consistent accuracy predictions.

Zhou et al. [2016] created a system to make a dynamic inventory forecast system. In this platform an ensemble algorithm is used, and then multiple factors that can influence inventory management system, such as seasonality and trends, are taken into account. The ensemble algorithm is made up of the algorithms: Linear Regression, Neural Network, Regression Tree, Gradient Boosting Regression Tree, Support Vector Machines and Gaussian Process. In this paper, a weighted linear combining scheme is used for combining the results. In this strategy, initially all algorithms start with the same weight, and are subsequently updated in each iteration, based on calculated metrics for each algorithm. The calculated metrics use mean absolute error, mean squared error and mean absolute percentage error. The data used in this study refers to the data of sales of LCDs (Liquid Crystal Display) and plasma TVs of one of the largest TV retailers in China. The results were quite satisfactory, showing much more accuracy than traditional forecasting methods such as Moving Average and Same Period Comparison.

2.3 Comparison of Forecasting Techniques

Bon and Ng [2017], with the aim of optimizing the prediction needs of a specific hospital product, used ten algorithms, all of them based on statistical methods, to make predictions of data and evaluate their performance. The data from the study were related to the use of the drug *Panadol 650mg*, data collected from the *Tun Hussein Onn Malaysia* university hospital. Sixty eight months of data were used, from January 2012 to August 2017. The authors applied the following algorithms to this data: Single moving average, Single exponential smoothing, Double moving average, Double exponential smoothing, Regression, Holt-Winter's additive, Seasonal additive, Holt-Winter's multiplicative, Seasonal multiplicative and ARIMA. The evaluation metrics of the algorithms employed was the Root Mean Square Error (RMSE). On analyzing the results, the regression analysis algorithm registered the smaller error in the predictions. However, this study has a limitation, it only performs forecasting of the needs of one product.

It is important not only to understand the behaviour and results of the application of statistical approaches in solving prediction problems but also to evaluate the performance of machine learning techniques and draw conclusions.

It was with this objective in mind that Carbonneau et al. [2008] applied a comparative analysis between advanced methods of machine learning and traditional methods of forecasting. Since the set of data on which one is working influences the results obtained two different sets of data were used. The first relates to data collected from a simulation of a supply chain. The second set of data relates to actual data from orders originating from the Canadian foundry industry. The algorithms of machine learning used were Neural Networks, SVM and Recurrent Neural Networks (RNN). On the statistical methods side are Naive Forecast, Average, Moving Average, Trend and Multiple Linear Regression. The comparison measurement of the performance of the algorithms used was Mean Average Error (MAE). The authors were convinced that the results of machine learning techniques would exceed the results of statistical methods. This belief was based on the fact that machine learning techniques were able to generate nonlinear models that would supposedly make better approximations of the complex behavioural patterns of the data than linear models. The results showed that, for the set of simulated data, the more elaborate techniques implemented in this study presented the best results. However, the improvement regarding the statistical method that showed the best results, Multiple Linear Regression, was not very significant. Yet, in the set of actual data, the algorithms Support Vector Machines and Recurrent Neural Networks were the ones which presented the lowest levels of error.

Another study, carried out by Gaur et al. [2015], assessed the behaviour of two techniques of machine learning for forecasting demand in a supply chain. The methodology used can be divided in two phases. In a first phase, the algorithms k-Nearest Neighbour (KNN) and Bayesian Networks (BN) were used for a set of data consisting of 1200 tuples with 35 attributes. This data was initially processed, where exceptions were removed and missing values were filled in. In a second phase, for the same data set, the AdaBoos algorithm was use to improve the performance of the algorithms implemented in the first phase of the study. The conclusion the authors came to was that the BN algorithm improved the performance when applied to the AdaBoost technique, having better accuracy than the KNN algorithm.

According to Chu and Zhang [2003], having large sets of data does not necessarily mean an improvement in predictive performance. In his study, the objective is to compare the predictive accuracy of retail sales performance between linear and nonlinear models. On the linear models, three algorithms were used: ARIMA, regression with a dummy variable and regression with trigonometric variables. The algorithm used for nonlinear models was neural networks. The results suggest that the best predictive approach was the use of neural networks using data without seasonality.

2.4 Hybrid Approaches

The use of an approach with just one algorithm does not work well in all situations. So Hybrid approaches emerge, which can be a good strategy, since they have the ability to model both linear and non-linear data [Zhang, 2003]. The combinations of ARIMA and ANN models which have been recently used, have been showing systematic improvements in performance [Zhang, 2003, Aslanargun et al., 2007, Aburto and Weber, 2007, Jain and Kumar, 2007, Díaz-Robles et al., 2008, Khashei and Bijari, 2010, Wang et al., 2013].

Zhang [2003] sees the usefulness of this method to, in a first phase, using the ARIMA algorithm, study the linear component of the problem. In a second phase, a neural network is built to analyze the information on the non-linearity of the data. The results obtained are better if the hybrid approach is used instead of using the algorithms separately for the same problem.

Khashei and Bijari [2010], in the same way, first resorted to the ARIMA algorithm to carry out some data processing, which will subsequently be used by a neural network. The neuronal network will generate a model capturing information included in the data, being this model used in forecasting. According to the author, this approach is a good alternative to approaches that only use neuronal networks when it is essential to obtain high levels of accuracy.

2.5 Statistical Methods vs Machine Learning

There are two aims when analyzing data. We may want to make predictions, where we want to be able to predict the response of a system given a specific input. The other aim is to extract information from the relation of certain variables. Data modelling based on statistical methods assumes that the data is generated by means of deterministic data models. These models are parametric since they assume a set of finite parameters to formalize the relationships of variables in mathematical equations. We can classify these methods as white box because the formal specification of the model is known.

Machine learning methods are non-parametric because it is assumed that the distribution of the data does not follow previously defined premises. They can be defined as black box methods precisely because they abstract the construction of the user model. This approach focuses more on research and identification of patterns in a set of data. It can, for example, be applied to complex data sets with large quantity of data and attributes [Breiman, 2001]. Image recognition, nonlinear time series forecasts, financial market forecasts, among others, are example of application areas where statistical approaches are not applicable and where machine learning is preferable.

Table 2.1 identifies some advantages and disadvantages of some prediction algorithms presented.

2.6 Accuracy Measures

Some accuracy metrics can be computed in order to evaluate the performance of a regression model.

2.6.1 Root Mean Squared Error

It is an often used metric which measures the average magnitude of the error. It is the square root of the average of squared differences between prediction and actual observation. *RMSE* is defined as:

$$RMSE = \sqrt{\frac{1}{n}\sum_{t=1}^{n}(y_t - \hat{y}_t)^2} \tag{2.13}$$

where y_t is the actual observation and $\hat{y}_t$ is the forecast produced by the model at point t.

2.6.2 Normalized Root Mean Squared Error

To facilitate the comparison of *RMSE* between data with different scales, *NRMSE* has been computed. It is defined as:

$$NRMSE = \frac{RMSE}{\bar{y}} \tag{2.14}$$

where $\bar{y}$ is the mean of the measured data.

Table 2.1: Advantages and disadvantages of forecast algorithms.

Algorithm	**Advantages**	**Disadvantages**
Linear Regression	Works for sets of any size. Simple and intuitive to use and understand.	Models only linear relationships. Sensitive to the presence of outliers.
Triple Exponential Smoothing	Able to handle sets of data with trends and seasonality.	Inability to establish relationship between several variables. Models only linear relationships.
ARIMA	Able to handle sets of data with trends and seasonality. Good predictions in the short term.	Needs large sets of data (at least 50 observations) Inability to establish relationship between several variables. Models only linear relationships.
Neural Networks	Able to represent complex and non-linear standards. Good predictive capacity.	Does not deal with missing values. Data must be standardized. Needs hyper-parameters tuning. Slow in the training phase. Difficult to interpret.
SVM	Effective for data with many dimensions. Able to represent complex and non-linear patterns. Good predictive capacity.	Does not deal with missing values. Needs hyper-parameters tuning. Slow in the training phase.
Random Forest	Good generalization capacity. Good predictive capacity	Difficult to interpret. Poor performance on imbalanced data.

2.6.3 Mean Absolute Percentage Error

It is scale-independent metric which is computed through a term-by-term comparison of the relative error in the prediction with respect to the actual value of the variable. It is defined as:

$$MAPE = \frac{1}{n}\sum_{t=1}^{n}\left|\frac{y_t - \hat{y}_t}{y_t}\right| * 100 \tag{2.15}$$

where y_t is the actual observation and $\hat{y}_t$ is the forecast produced by the model at point t.

2.6.4 Mean Arctangent Absolute Percentage Error

MAPE has a significant disadvantage because it produces infinite or undefined values when the actual values are zero or close to zero. To solve this problem, Kim and Kim [2016] had proposed a new metric using arctangent. MAAPE is defined as:

$$MAAPE = \frac{1}{n}\sum_{t=1}^{n}(AAPE_t) \tag{2.16}$$

where

$$AAPE_t = arctan\left(\left|\frac{y_t - \hat{y}_t}{y_t}\right|\right) \tag{2.17}$$

The corresponding range of AAPE is $\left[0, \frac{\pi}{2}\right]$

Chapter 3

Methodology

The methodology followed started with an exploratory analysis of the data, where all items have been categorized according to their demand pattern. Afterwards, some steps, namely data preparation and data splitting, have been carried out before proceeding to model calibration. The models developed were used to perform demand forecast. After all, accuracy metrics were collected and analyzed. The diagram shown in Figure 3.1 summarizes the approach.

All the work was developed using R language [R Core Team, 2012]. This language has been selected because it is open source. For this reason, there is a huge community of developers, which is a great advantage for learning R programming. The R language is also a powerful tool for statistical analysis, visualization and for machine learning approaches, since it has the support of many packages available in the *CRAN* repository.

3.1 Demand Patterns

The quality of a forecast may strongly depend on the demand history characteristics. To facilitate inventory management it is necessary to classify the items according to their consumption patterns. This classification is also important to infer a consumption pattern for products with no history. We followed the categorization framework proposed by Boylan et al. [2008]. The items can be classified as smooth, lumpy, intermittent and erratic as shown in Figure 3.2.

The two coefficients used in this classification are the average demand interval (p) and square of the coefficient of variation (CV^2). The average demand interval measures the regularity of a demand in time while the square of the coefficient of variation measures the variation in the demand quantities.

A smooth pattern is described by a regular demand in time and quantity. When the demand has regular occurrences in time but the quantity variation is high we are dealing with a erratic demand. Intermittent demand is characterized by high variation in the interval between two demands and

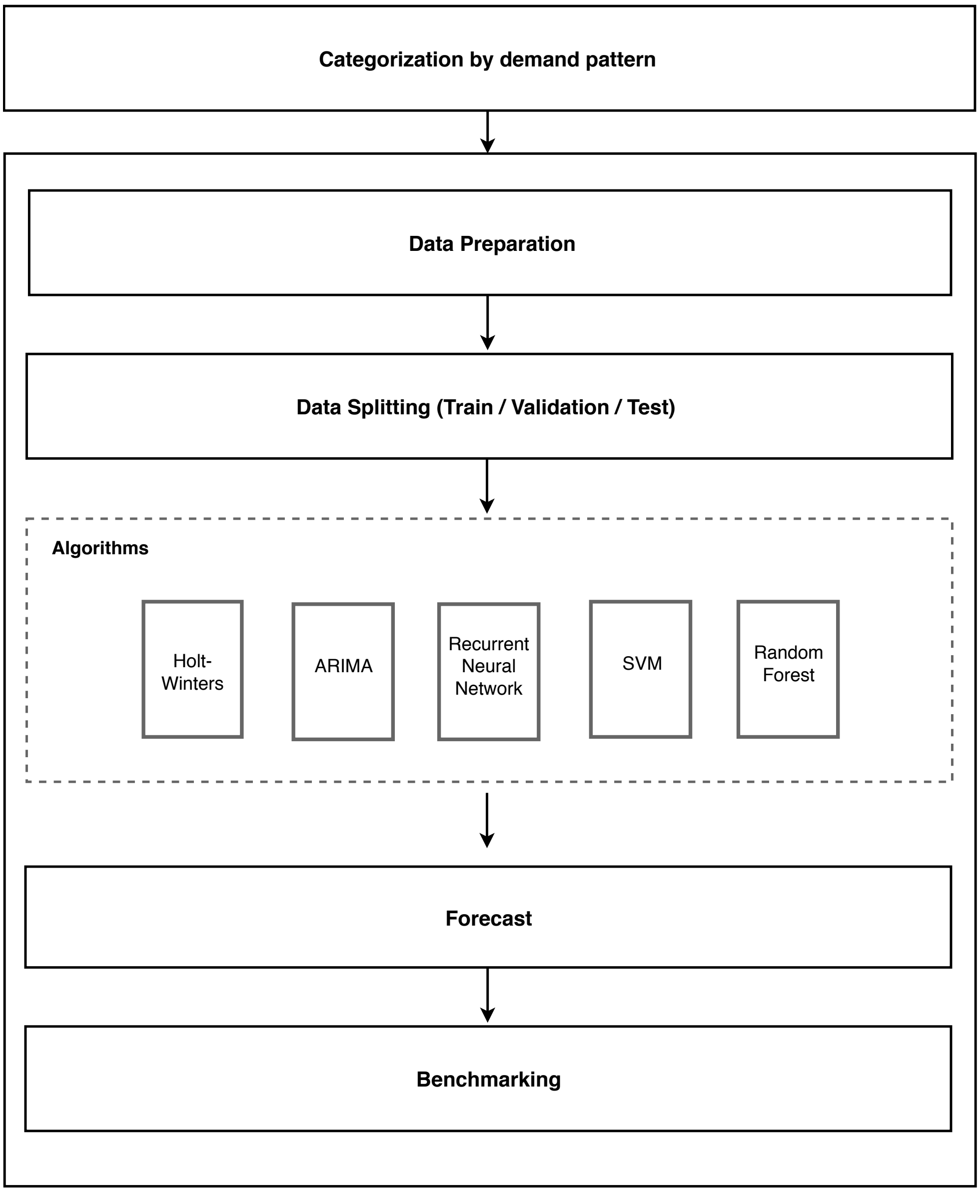


Figure 3.1: Schematic diagram of the research method

a low variation in demand quantity. Lumpy demand has a large variation in the demand interval between two demands and quantity. Figure 3.3 illustrates the demand patterns identified.

3.2 Data set

The data used to perform this analysis is related to the consumption history of a large urban multi-specialty Portuguese hospital, Centro Hospitalar de Vila Nova de Gaia (CHVNG). This data set contains data collected between 2015 and 2017 and it is composed of discrete variables which only records consumptions of medical consumables and does not register stock outs. All the items

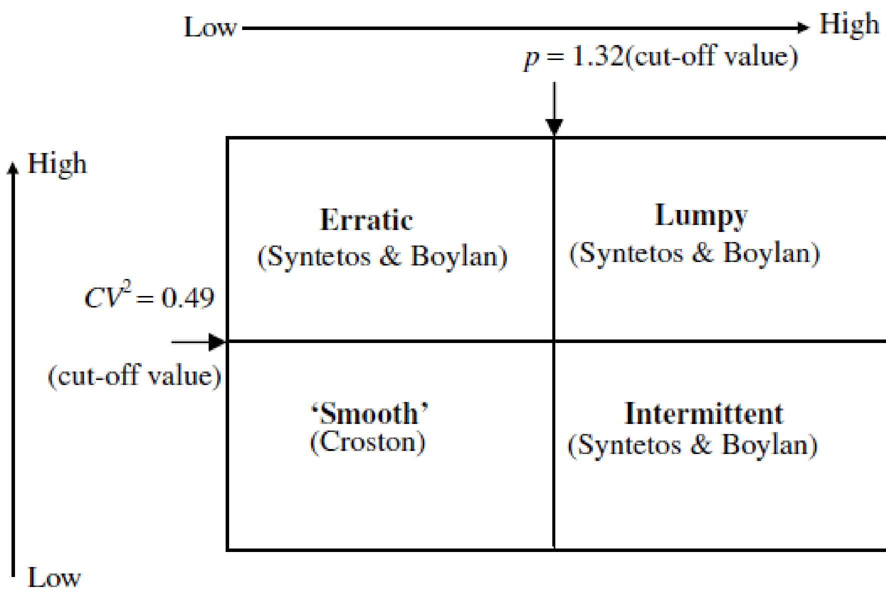


Figure 3.2: Categorization scheme [Boylan et al., 2008]

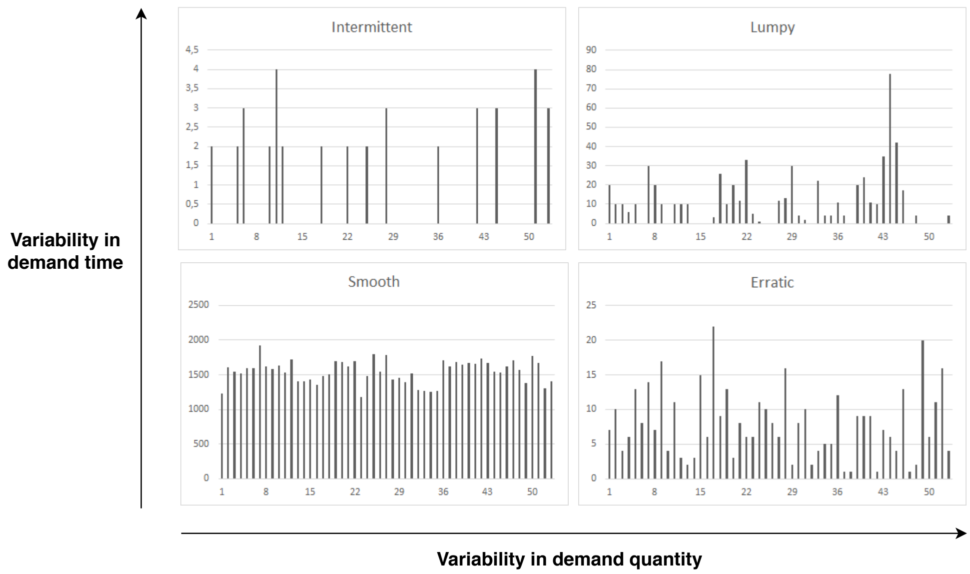


Figure 3.3: Graphical representation of demand patterns

present in the data set were classified using the classification explained above. Ten products of each category have been selected. This choice falls on medicines and important products used in medical treatments. Table 3.1 represents the products in more detail.

Table 3.1: Summary product information

Product no.	Name	CV^2	p	Demand type
2275655	Monofilamentar de Polipropileno	0.58	1.01	erratic
2344537	Tampa Para Sonda Nasogastrica	0.05	1.01	smooth
2349004	Filtro Anti-Bacteriano para Injecção	1.62	0.18	intermittent
2349984	Conector com membrana porta-injecção	1.33	1.47	lumpy
2999925	Tubo Cristal	0.09	1.01	smooth
9911315	Anticorpo Pre-diluido	0.61	1.45	lumpy
9931291	Mioglobina	0.33	2.01	intermittent
9931296	CA 15.3	0.46	3.00	intermittent
9931301	Citomegalovirus IGG	0.10	2.69	intermittent
9931352	Magnesio	0.21	2.72	intermittent
110416009	Eritromicina 250 mg/5 ml	0.39	1.41	intermittent
110416030	Amoxicilina 500 mg	0.60	1.12	erratic
110416252	Cefuroxima 500 mg	0.57	1.33	lumpy
110440249	Solução antisséptica etanol + Propanol 500 ml	0.63	1.01	erratic
110808065	Lidocaína 40 mg/g	0.25	3.67	intermittent
110808186	Bupivacaína 50 mg/10 ml	1.21	1.37	lumpy
110820153	Acido valpróico 300 mg	0.76	1.41	lumpy
110832025	Clobazam 10 mg	0.56	1.64	lumpy
110832500	Tetrabenazina 25 mg	0.23	1.41	intermittent
110844010	Acido acetilsalicílico 100 mg	0.03	1.01	smooth
110844214	Paracetamol 10 mg/ml	0.04	1.01	smooth
111620050	Dinitrato de isossorbida 5 mg	0.77	1.23	erratic
112012248	Proteínas coagulantes 500 U.I.	0.53	1.13	erratic
112408010	Aminofilina 240 mg/10 ml	0.72	1.02	erratic
113612005	Bromocriptina 2.5 mg	0.90	1.60	lumpy
114004090	Diclofenac 50 mg	0.17	1.01	smooth
114004160	Ibuprofeno 400 mg	0.05	1.01	smooth
114804045	Paricalcitol 1 μg	0.57	2.09	lumpy
114804114	Carbonato de cálcio 500 mg	0.63	1.05	erratic
114808048	Dieta Lactea De Formula Adaptada	0.52	1.03	erratic
115204851	Bicarbonato de sódio 1000 mg	0.09	1.01	smooth
116008160	Fenilefrina 2.5 mg/ml	0.54	1.05	erratic
116804072	Dasatinib 100 mg	0.14	2.74	intermittent
116804182	Docetaxel 20 mg/ml	0.54	1.23	erratic
116804310	Micofenolato de mofetil 250 mg	0.77	2.16	lumpy
116804311	Micofenolato de mofetil 500 mg	0.19	1.01	smooth
117204048	Acetilcisteína 600 mg	0.15	1.01	smooth
117608391	Fluoresceína 2.5 mg/ml + Oxibuprocaína 4 mg/ml	0.24	1.43	intermittent
118404206	Pamidronato de sódio 3 mg/ml	0.51	4.94	lumpy
118404427	Solução para preparações injectáveis	0.04	1.01	smooth

3.2.1 Data Preparation

Initially, the data set had the consumption history data grouped by day. Data was grouped weekly in order to reduce some noise in data and because it follows the ordering of the products.

For the statistical algorithms approach, the data was transformed into a time series object.

On the other hand, machine learning algorithms cannot handle time series object. For that reason, it has been mandatory to define an input structure for the models. In this structure, the features of the model are lagged variables which represents previous timesteps. Varying the number of previous timesteps selected it is possible to define different input structures. Table 3.2 shows the input structures of the models defined for forecasting time series.

Table 3.2: The input structure of the models for forecasting of time series

Model	Input Structure
M1	$x_t = f(x_{t-1}, x_{t-2}, x_{t-3})$
M2	$x_t = f(x_{t-1}, x_{t-2}, x_{t-3}, x_{t-4}, x_{t-5})$
M3	$x_t = f(x_{t-1}, x_{t-2}, x_{t-3}, x_{t-4}, x_{t-5}, x_{t-6}, x_{t-7}, x_{t-8})$

Regarding to neural networks, it is standard practice to normalize the inputs before applying them. A z-score normalization (Eq 3.1) is applied to input variables.

$$x_{new} = \frac{x - \mu}{\sigma} \quad , \tag{3.1}$$

where μ is the mean of the population and σ is the standard deviation of the population.

In this way, the network output always falls into a normalized range. The network output is then transformed back into the original data units.

3.2.2 Data Splitting

Data splitting it is an important step to have a reliable estimation of the model performance. The data has been divided in two different datasets: the training data and the test data. The training data is used to train the model and come up with a predictor. The test data is used to estimate the predictive accuracy of the model in unseen data. In some cases a validation set is created to optimize the parameters of the model. This division is extremely important because, at the end of training process, the model should be able to generalize well to new data.

As our dataset is composed of data collected over three years, the training set used corresponds to the first two years and the test set to the last year. In the healthcare industry, seasonality plays an important role. It is essential to capture all the patterns and that is why the first two years were use to train.

3.3 Model Calibration

One of the key factors for the construction of a good predictive model has to do with hyper-parameter tuning. For each algorithm used, we describe both development and the hyper-parameter tuning phases.

3.3.1 Holt Winters

The Holt Winters algorithm was implemented using the R' stats package . This algorithm has 3 parameters to be defined, which have a range from zero to one. The optimizer will automatically define the value of these parameters in order to minimize the squared error of the predictions. By default, the optimizer uses the following starting values: $alpha = 0.3$, $beta = 0.1$, $gamma = 0.1$. Sometimes, it could be the case that the optimizer gets stuck at a local minimum. To overcome this problem, another Holt Winters model has been created changing the initial values of the optimizer to $alpha = 0.3$, $beta = 0.01$, $gamma = 0.01$. Since $beta$ and $gamma$ represents trend and seasonality components respectively, these values have been chosen for time series that does not show evidence of these components.

3.3.2 ARIMA

ARIMA algorithm was implemented using the forecast package [Hyndman et al., 2018]. The optimizer will look at an autocorrelation on the data to define the Moving Average (MA) model, a partial autocorrelation of the data to define the Autoregressive (AR) model and an extended autocorrelation of the data to combine AR and MA models. If the data is non-stationary, we will have the Integrative part. This represents how many times the series is differenced.

The three parameters that minimize the Schwartz Bayesian Information Criterion (BIC) have been chosen.

3.3.3 Recurrent Neural Network

A classic three-layer recurrent neural network has been implemented. The choice of recurrent neural network was due to the fact it exploits the temporal ordering of data points. This is a sequential network with one input layer, one hidden layer with gated recurrent units and one output layer. The activation function and optmizer used was hyperbolic tangent and *rmsprop* respectively. The mean squared error (MSE) was used as the loss function. This choice is justified by the fact that it is a quadratic function, which means that its derivative is defined.

The network has been trained for 5000, 2500 and 500 epochs using a learning rate of 0.001. To test the ideal number of neurons in the hidden layer, the combinations $I/2$, I, $2I$ and $2I+1$, where I is the number of features, have been selected.

Due to high computational costs, cross validation has not been performed for choosing the best architectural structure and hyper-parameters setting. Instead, a validation set has been created to evaluate the model performance and avoid overfitting. We have used grid search technique to select the best model. The network that yielded the best results has been selected for forecasting.

These RNN have been implemented using the Keras API, [Allaire and Chollet, 2018], among with the TensorFlow framework. [Allaire and Tang, 2018]

3.3.4 SVM

It is known that SVM performance relies on hyper-parameters tuning and kernel choice. Radial basis function (RBF) kernel was used for its good performances in time series forecasting [Samsudin et al., 2010, Wang et al., 2009].

In order to determine a good value for hyper-parameters C and *sigma* was used a grid search where the ranges of search has been set to $[1,15]$ at increment of 1.0 for C and $[0.25,5]$ at increment of 0.25 for *sigma*. For each combination, a 10-fold cross validation on the training set, repeated ten times, has been used to increase the confidence of the results. The metric used to determine the best settings was RMSE. The model that yielded the best results has been selected for forecasting.

SVM has been implemented using *caret* package, [from Jed Wing et al., 2018], making use of *kernlab* package [Karatzoglou et al., 2004].

3.3.5 Random Forest

Similar to what has been done in SVM's implementation, a grid search has been use to define the best value of *mtry* parameter, number of variables randomly sampled as candidates at each split. The range search has been set to $[1,I]$, where I is the number of features, at increment of 1.0 and the number of trees to grow being fixed to 500. A 10-fold cross validation on the training set, repeated ten times, has been used with RMSE as metric. The model that yielded the best results has been selected for forecasting.

Random Forest has been implemented using *caret* package, [from Jed Wing et al., 2018], making use of *randomForest* package [Liaw and Wiener, 2002].

3.4 Model performance evaluation

In order to evaluate the performance of the proposed algorithms, some accuracy measures have been computed. RMSE is a useful metric because larger errors are more penalized. This is a good metric if we want to keep the magnitude of the errors under control, rather than the number of errors Although this a scale-dependent metric. RMSE of items with different scales are not comparable. NRMSE is a non-dimensional form of RMSE and it is appropriate to compare RMSE with different units.

MAPE is other widely used forecast measure accuracy. This is a percentage error which computes the absolute percentage error of forecasts. MAPE does not treat positive and negative variations equally. It places heavier penalties on positive errors than on negative errors [Davydenko and Fildes, 2013]. This accuracy measure has some limitations which have been overcome by MAAPE. It can produce infinite or undefined values when the actual values are zero or close to zero [Kim and Kim, 2016].

When the time comes to decide which algorithm to use, computational complexity is an extremely important factor. The average execution times of each algorithm have been computed to support the decision-making.

Chapter 4

Results

In this chapter, the results obtained in this dissertation will be presented. Using the methodology described in Chapter 3, for each algorithm implemented the model that yield the best results has been selected to forecast. The forecast horizon chosen was fifty weeks, approximately one year.

Firstly, the results will be presented in a aggregated form and then in a more detailed view for each category. To analyze the results inter-category, NRMSE and MAAPE will be used as long as they are scale-independent metrics. To analyze the results intra-category, all metrics, when applicable, will be interpreted. Lastly, the average of execution times of each algorithm will be presented.

4.1 Aggregated results

Regarding the MAAPE metric, Figure 4.1 shows an overview of the algorithms' performance, using MAAPE, for all categories. In this figure, the results of each algorithm for all products are presented, grouped by category. As expected, the category where the best results are achieved is "Smooth". This could be explained by the fact this demand type is the most regular one. It has strong seasonality and trend components.

Lumpy and intermittent demands have relatively bad results with high dispersion. The fact that these type of demand have high variability in demand timing may not be good for these kind of forecast algorithms.

With regards to the erratic category, the results remain unsatisfactory despite being better than the majority of lumpy's and intermittent's.

Table 4.1 shows how many times a certain algorithm performs better than others for the same product. MAAPE was used in this analysis. Machine learning algorithms outperform statistical methods for items which exhibit a smooth demand pattern. For the other demand patterns, there is not a clear distinction of which type of approach to follow. It should be pointed out that the HW algorithm has never outperformed the other alternatives.

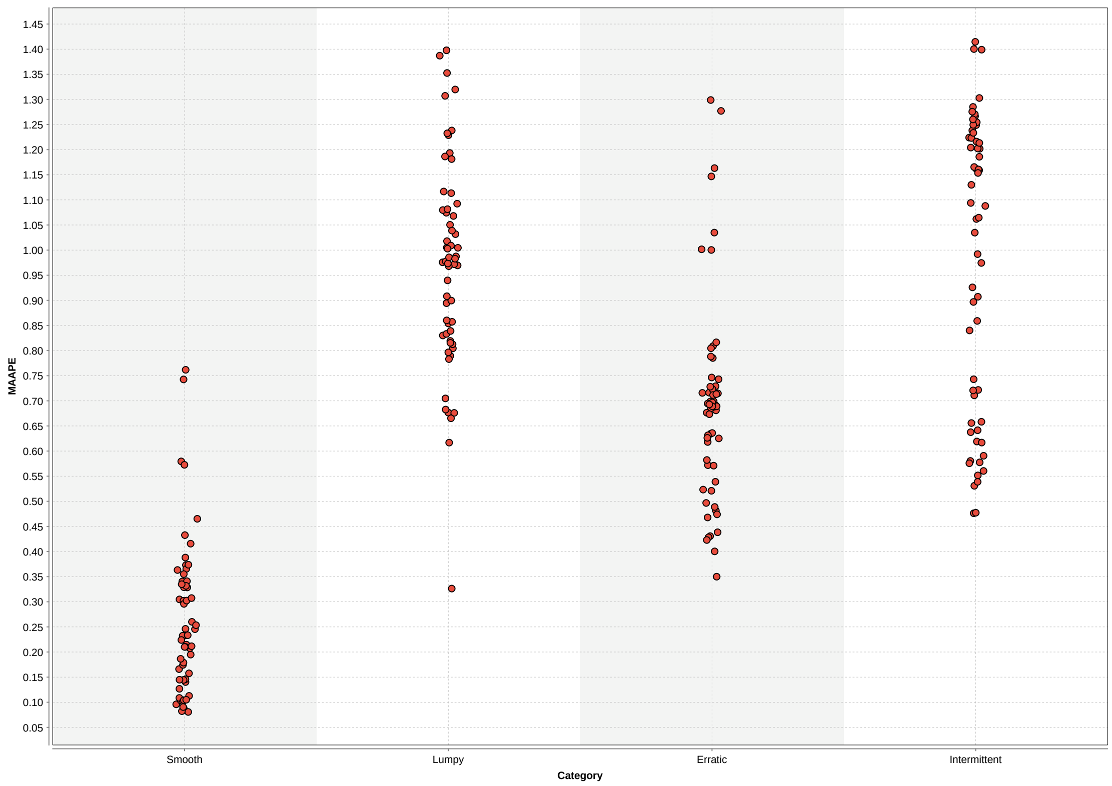


Figure 4.1: Aggregated results for MAAPE metric

Table 4.1: Winning rank of algorithms using MAAPE

Category	HW	ARIMA	RNN	SVM	RF	
Smooth	0	0	3	4	3	10
Lumpy	0	4	4	1	1	10
Erratic	0	0	2	7	1	10
Intermittent	0	4	1	5	0	10
	0	8	10	17	5	40

Similar to what was done with MAAPE, the results for NRMSE will now be presented. Figure 4.2 shows the algorithms' performance for NRMSE metric. The results are quite similar. We can see that forecast for items with smooth demand pattern are the ones with lower root mean squared error. The other three categories present worse results, with some outliers and high dispersion.

Table 4.2 shows how many times a certain algorithm performs better than the others for the same product using NRMSE. HW algorithm continues to never win and ARIMA, with this metric, does not win a single item. RF algorithm, contrarily to what happens with MAAPE, is better optimizing RMSE of forecasts. The other two machine learning algorithms seem to be a good choice when the objective is minimize RMSE. On the statistical methods side, the baseline algorithm is the one which is able to win in some items.

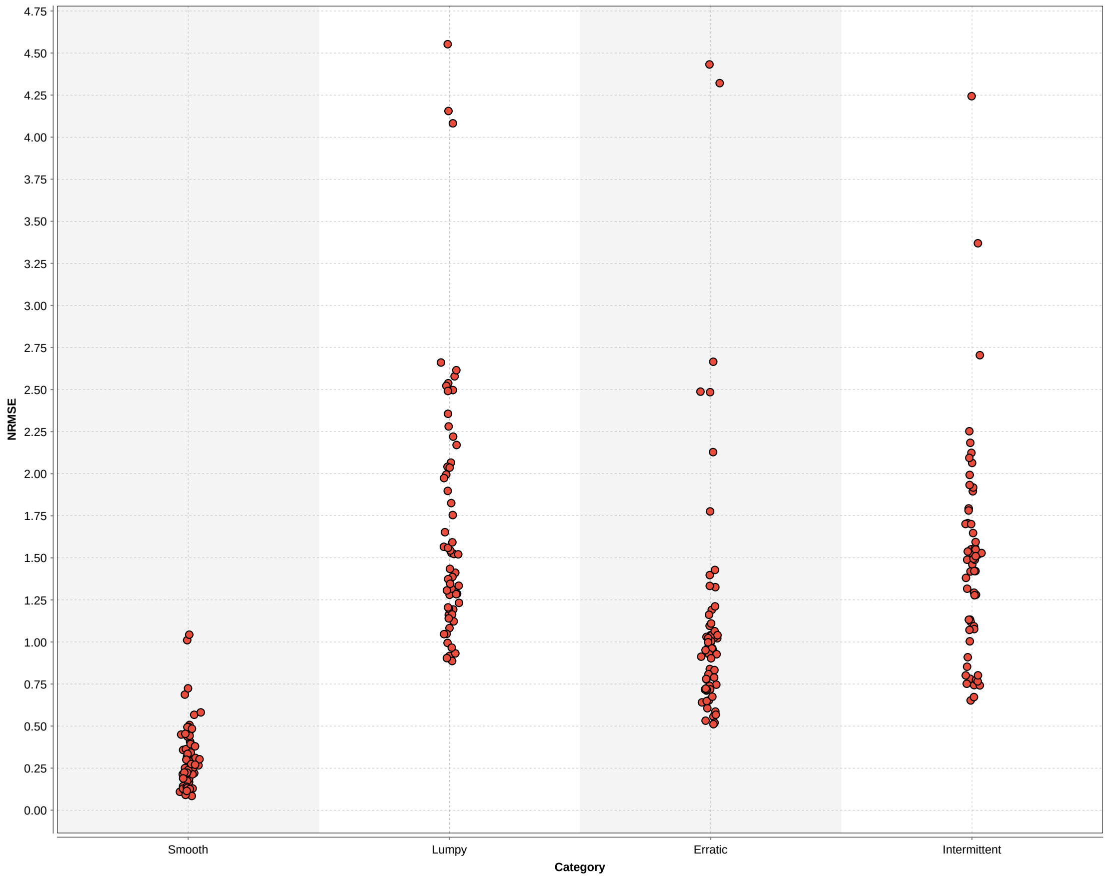


Figure 4.2: Aggregated results for NRMSE metric

Table 4.2: Winning rank of algorithms using NRMSE

Category	HW	ARIMA	RNN	SVM	RF	
Smooth	0	0	2	4	4	10
Lumpy	0	0	4	0	6	10
Erratic	0	0	6	2	2	10
Intermittent	0	0	2	4	4	10
	0	0	14	10	16	40

4.2 Smooth Category Results

In this section we will have a closer look to smooth category. Starting with MAAPE, we can see in Figure 4.3 that machine learning approach is the best. Analyzing Table 4.3, which has a statistics representation of both MAAPE and NRMSE metrics, we can conclude that Random Forest is the best algorithm to forecast smooth items. It is the one with lower mean and standard deviation. Statistical algorithms are not so stable as machine learning ones because their standard deviation is higher. These values achieved are similar with the ones in literature where products with smooth demand pattern have been used. [Cheng et al., 2016, Syntetos et al., 2010]

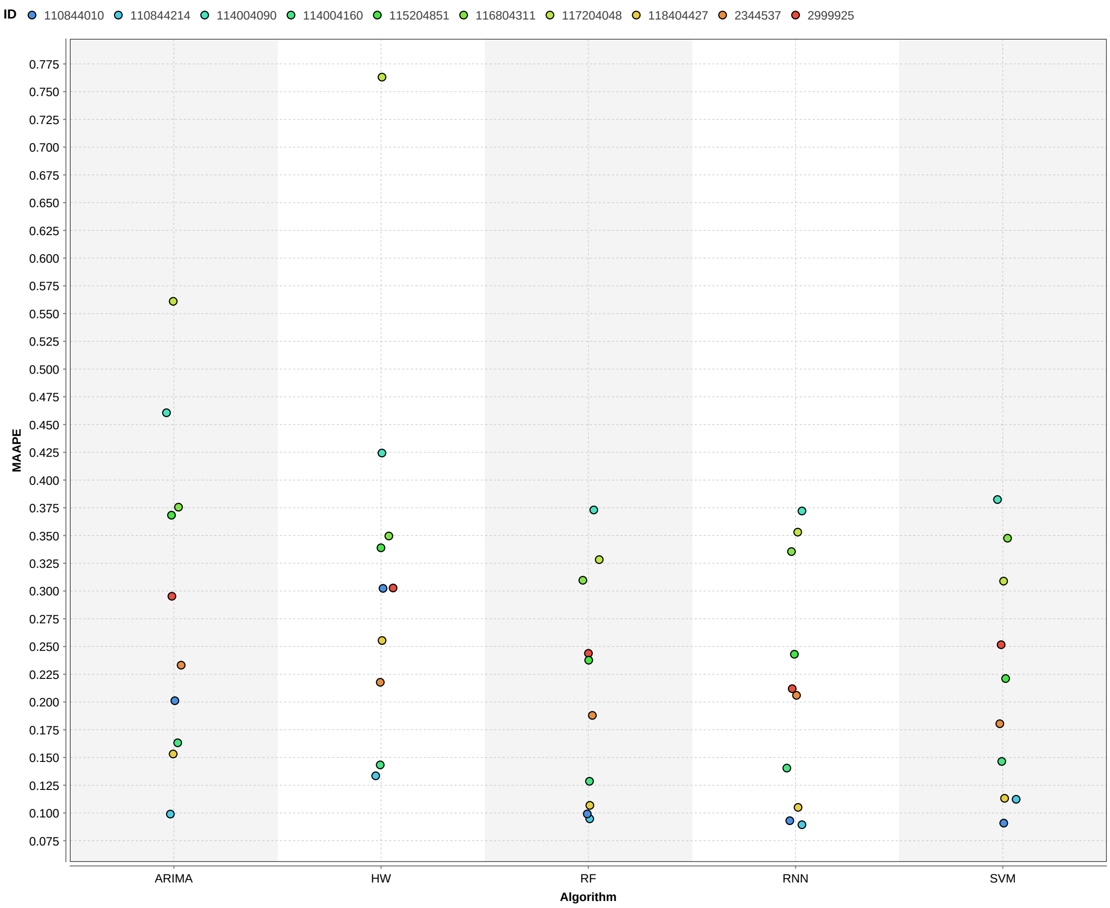


Figure 4.3: Performance of algorithms for Smooth Category using MAAPE

Table 4.3: MAAPE and NRMSE Statistics for Smooth Category

Algorithm	MAAPE					NRMSE				
	Mean	Std	CV	Max	Min	Mean	Std	CV	Max	Min
HW	0.322	0.180	0.558	0.767	0.136	0.409	0.261	0.638	1.061	0.159
ARIMA	0.292	0.149	0.509	0.563	0.103	0.366	0.187	0.511	0.693	0.128
RNN	0.216	0.108	0.499	0.372	0.092	0.273	0.140	0.513	0.503	0.123
SVM	0.216	0.105	0.485	0.380	0.089	0.265	0.130	0.492	0.475	0.113
RF	0.212	0.103	0.486	0.371	0.094	0.256	0.124	0.485	0.477	0.120

4.3 Lumpy Category Results

Analyzing both MAAPE and NRMSE erros, this is obviously a difficult category to forecast. Due to high variability demand in quantity and timing, it is quite impossible to have reliable forecast. ARIMA is the algorithm which has the lower MAAPE average. However, this is not the algorithm with lower standard deviation. SVM shows up as the algorithm with smallest standard deviation in the two metrics. Minimizing RMSE, Random Forest is the algorithm with smallest average.

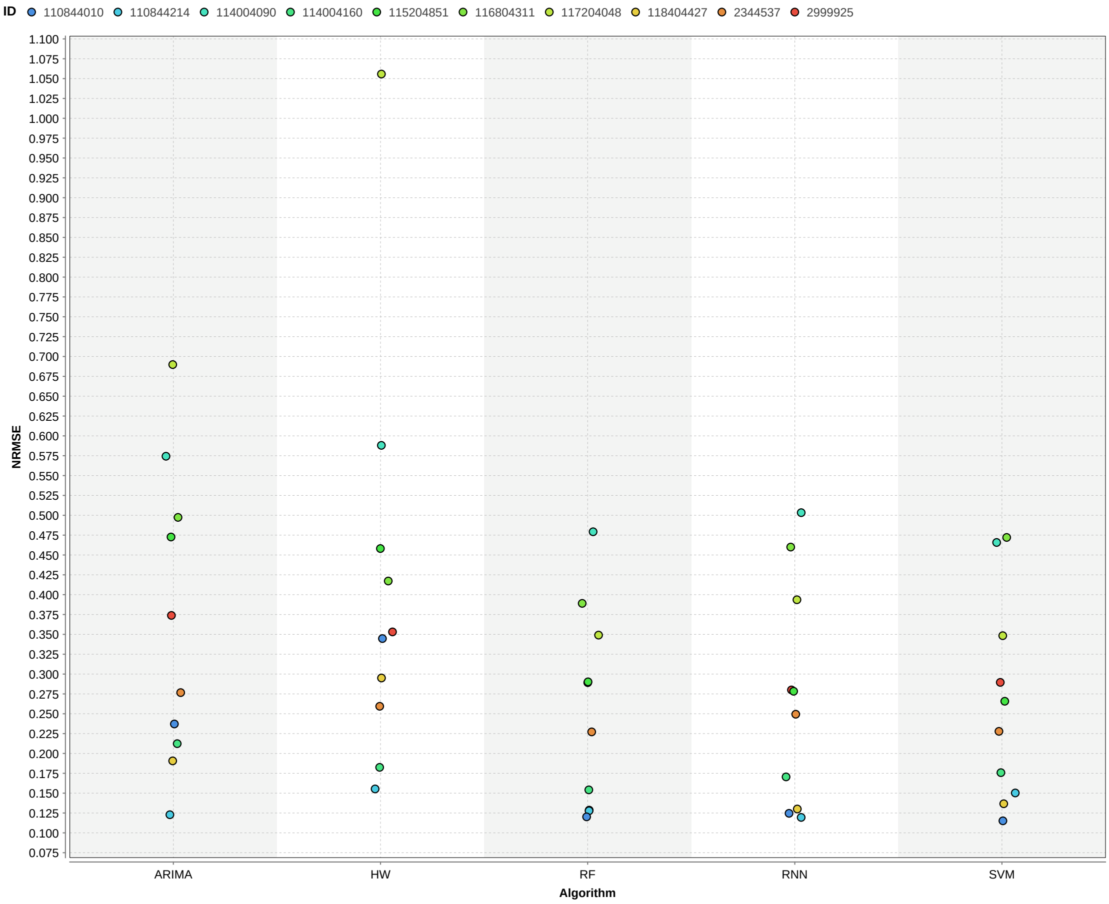


Figure 4.4: Performance of algorithms for Smooth Category using NRMSE

(Table 4.4). Globally, machine learning approach is the one that stands out when the objective is to minimize the RMSE (Figure 4.6).

Table 4.4: MAAPE and NRMSE Statistics for Lumpy Category

Algorithm	MAAPE					NRMSE				
	Mean	Std	CV	Max	Min	Mean	Std	CV	Max	Min
HW	1.051	0.199	0.189	1.398	0.702	2.225	1.047	0.471	4.457	1.039
ARIMA	0.885	0.232	0.262	1.179	0.336	2.054	0.874	0.426	4.116	1.178
RNN	0.962	0.194	0.201	1.329	0.672	1.434	0.524	0.365	2.646	0.918
SVM	0.942	0.186	0.197	1.350	0.676	1.455	0.491	0.338	2.529	0.923
RF	0.994	0.221	0.234	1.306	0.614	1.414	0.512	0.362	2.519	0.895

4.4 Erratic Category Results

The forecast errors for erratic demand continues considerably high as a result of its high variability in demand quantity. SVM is the algorithm with smallest MAAPE average, while RNN is the one

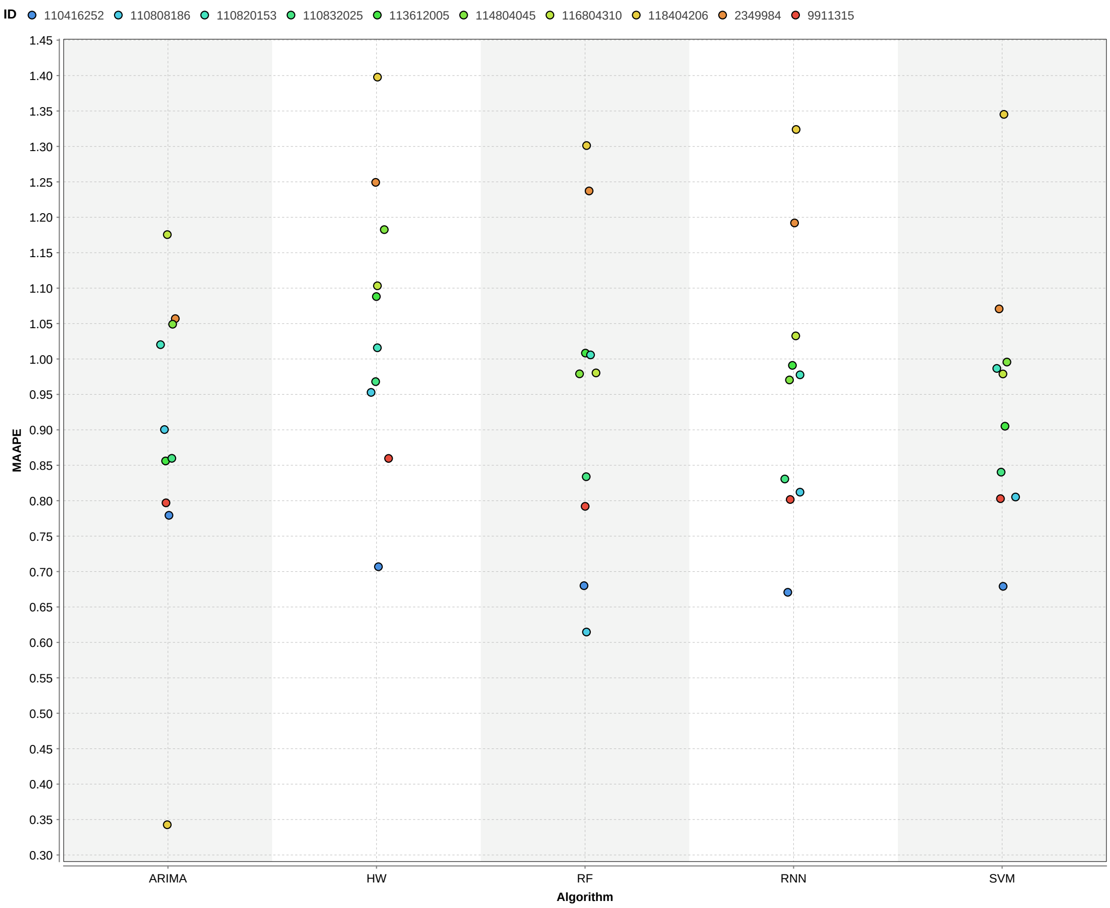


Figure 4.5: Performance of algorithms for Lumpy Category using MAAPE

with smallest standard variation. To minimize RMSE, RNN is the best choice for this demand type (Table 4.5).

Table 4.5: MAAPE and NRMSE Statistics for Erratic Category

Algorithm	MAAPE					NRMSE				
	Mean	Std	CV	Max	Min	Mean	Std	CV	Max	Min
HW	0.838	0.167	0.200	1.159	0.684	1.545	0.592	0.383	2.524	1.034
ARIMA	0.741	0.214	0.288	1.162	0.449	1.284	0.583	0.454	2.662	0.666
RNN	0.612	0.109	0.177	0.718	0.415	0.801	0.164	0.205	0.970	0.525
SVM	0.596	0.111	0.186	0.733	0.403	0.812	0.175	0.216	1.038	0.565
RF	0.620	0.113	0.182	0.785	0.422	0.804	0.173	0.216	1.040	0.536

4.5 Intermittent Category Results

Interpreting Table 4.6, we can see that the values of the metrics are considerably high. This is a hard category to forecast. When MAAPE is analyzed, ARIMA is the algorithm with smallest

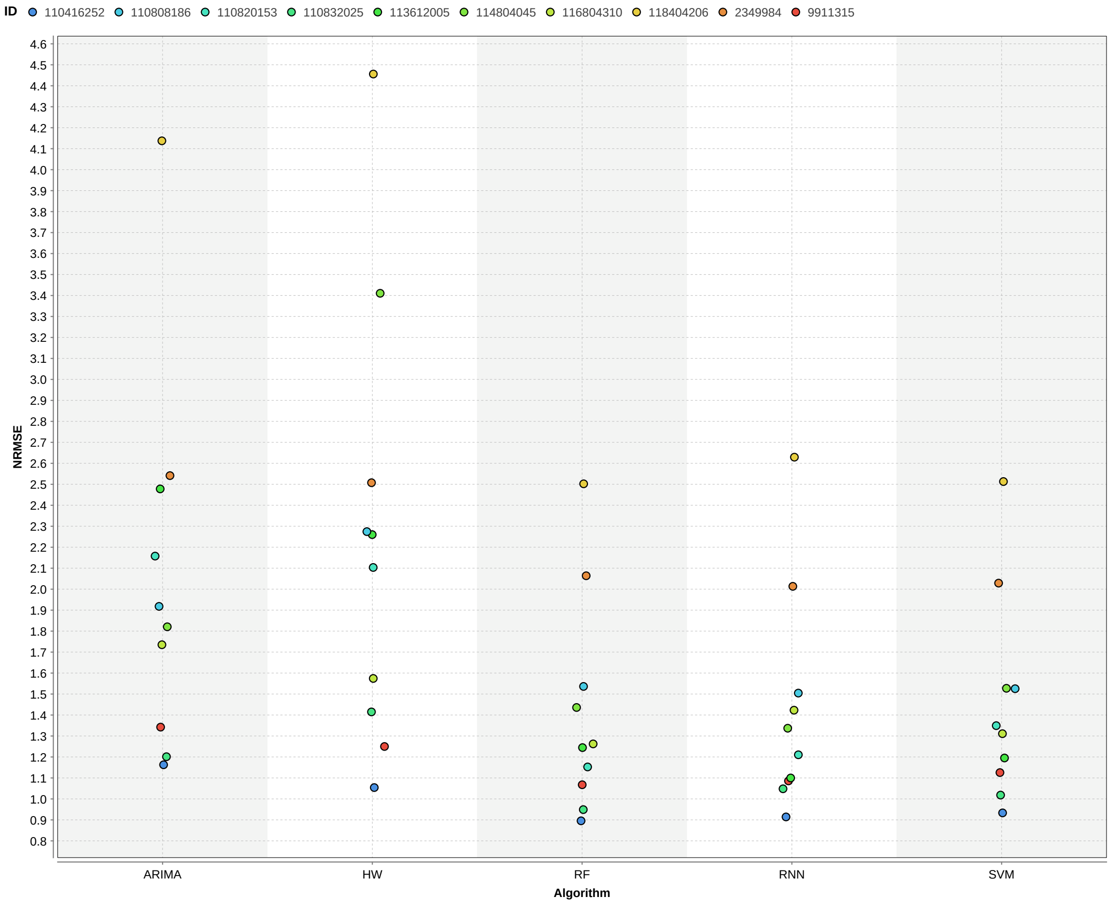


Figure 4.6: Performance of algorithms for Lumpy Category using NRMSE

average and Holt-Winters is the one with smallest standard deviation. On the other hand, machine learning algorithms prove to be the best choice to minimize RMSE. RNN is a good choice to forecast this type of demand.

Table 4.6: MAAPE and NRMSE Statistics for Intermittent Category

Algorithm	MAAPE					NRMSE				
	Mean	Std	CV	Max	Min	Mean	Std	CV	Max	Min
HW	1.078	0.242	0.225	1.414	0.715	1.889	0.857	0.454	4.120	1.086
ARIMA	0.840	0.311	0.371	1.391	0.487	1.898	0.717	0.378	3.344	1.072
RNN	0.991	0.285	0.287	1.304	0.553	1.256	0.343	0.273	1.665	0.716
SVM	0.971	0.305	0.314	1.265	0.534	1.275	0.431	0.338	1.903	0.662
RF	0.974	0.290	0.297	1.271	0.548	1.275	0.453	0.356	2.131	0.704

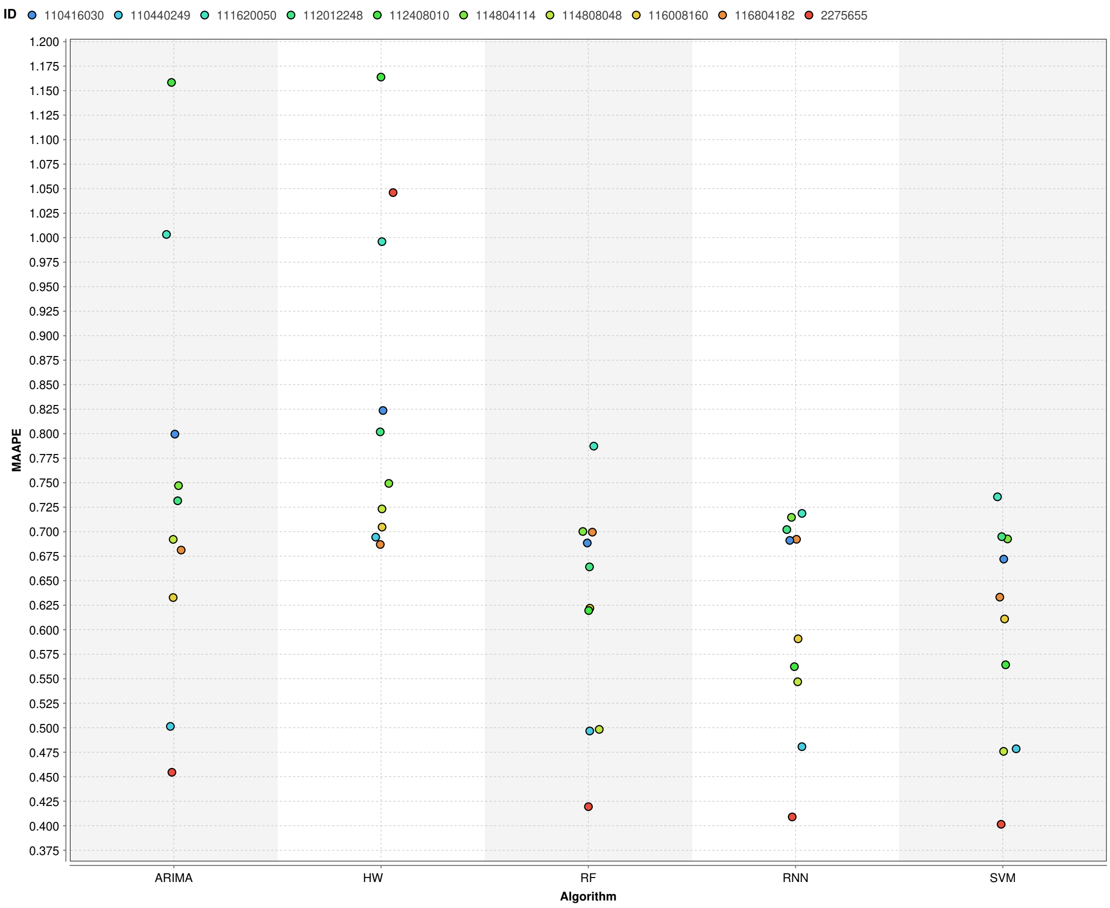


Figure 4.7: Performance of algorithms for Erratic Category using MAAPE

4.6 Execution Times Results

The information provided in Table 4.7 corresponds to the mean of the execution time of the algorithms during the training and testing phase. Computational time was estimated using a system with the following characteristics: Intel ®Xeon ®Processor E5-2650 @ 2.00 GHz and 128 GB RAM.

The training time corresponds to the time the algorithm takes to generate a model, while the testing time is the time to perform the forecast. Unsurprisingly, statistical algorithms have much lower execution time than machine learning algorithms. RNN is the algorithm that takes longer to run.

4.7 Results Discussion

As expected, the quality of forecast algorithms is strongly dependent of the demand category of the items. Ahmed et al. [2010], which conducted a large scale comparison study of different machine

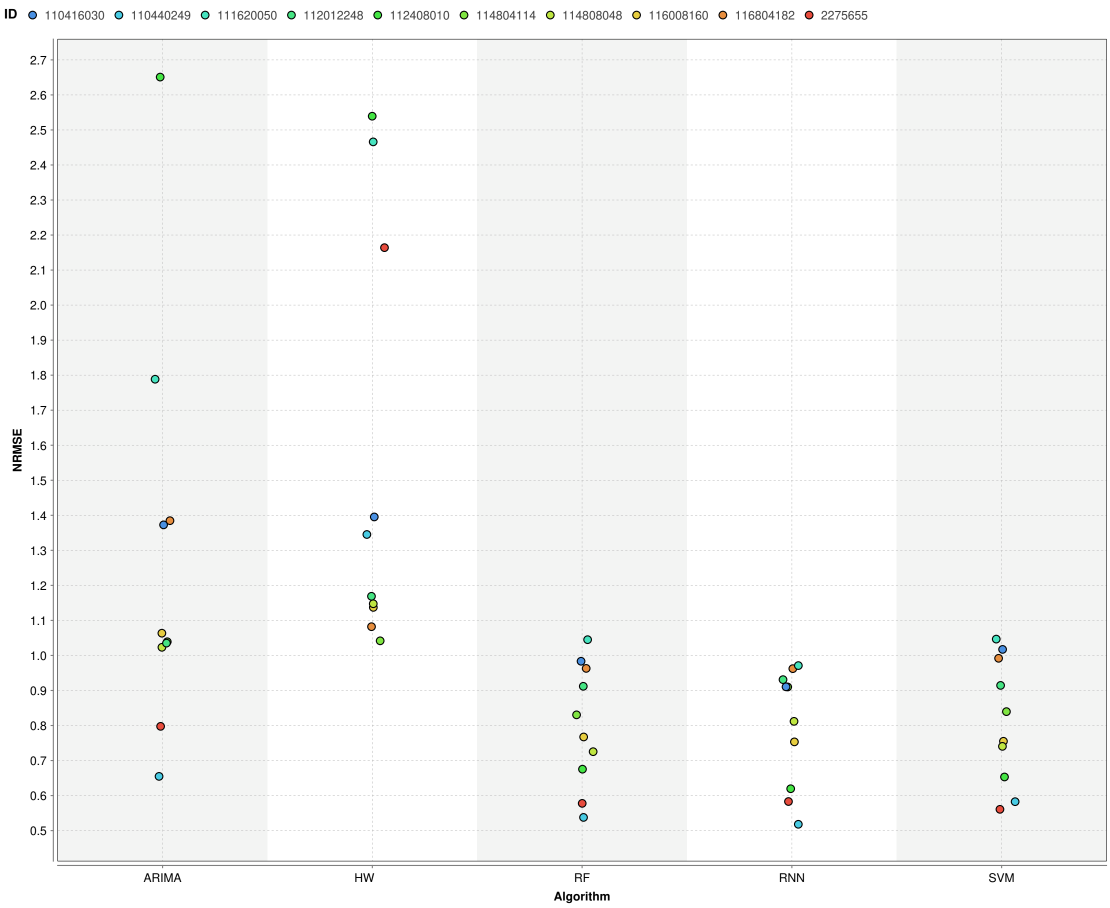


Figure 4.8: Performance of algorithms for Erratic Category using NRMSE

Table 4.7: Execution Times Results

Algorithm		Smooth	Lumpy	Erratic	Intermittent
HW	Train (s)	0.009	0.008	0.015	0.008
	Test (s)	0.107	0.093	0.091	0.093
ARIMA	Train (s)	0.121	0.1	0.079	0.193
	Test (s)	0.051	0.026	0.026	0.028
RNN	Train (s)	12420	11988	12636	11232
	Test (s)	0.54	0.84	1.43	1.62
SVM	Train (s)	322.69	314.14	317.30	306.60
	Test (s)	0.003	0.003	0.003	0.003
RF	Train (s)	51.46	49.39	52.30	37.44
	Test (s)	0.004	0.004	0.004	0.004

learning algorithms for time series forecast, concluded that certain time series' feature might favor one algorithm over the other. However, the results are not comparable to this study because neither the data used nor the metrics are the same.

The impact of randomness in the series is an important concern. It is a challenging problem

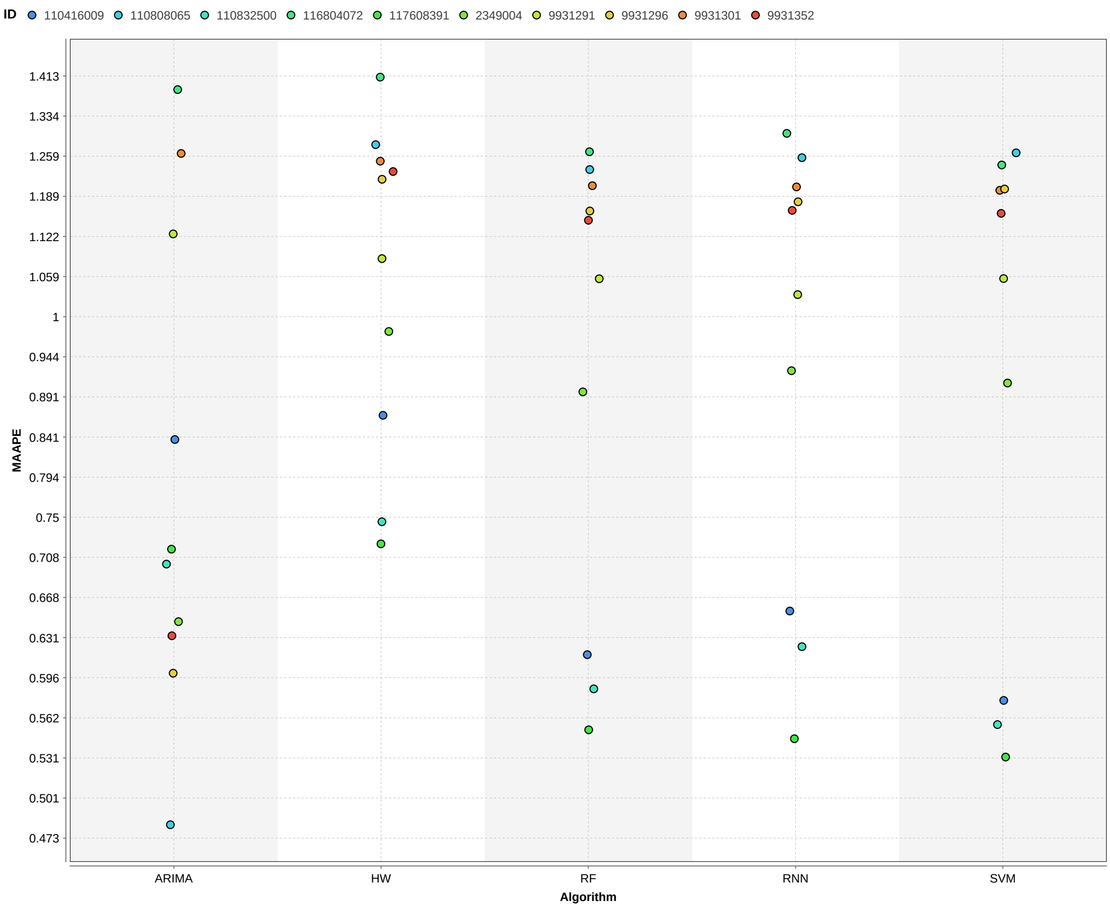


Figure 4.9: Performance of algorithms for Intermittent Category using MAAPE

for models to distinguish patterns from the noisy data. Since there is high variability in some categories characteristics, this could be an explanation for higher errors, especially for erratic, lumpy and intermittent.

It is important to define what metric is important in a hospital inventory management scenario. A stock out can be very problematic and shloud be avoided. Probably, it's better to have several errors with small amplitude instead of having one error with high amplitude. As RMSE is more useful when larger errors are undesirable, machine learning approach is the best option. For all categories, the worst machine learning algorithm is still better than the best statistical algorithm.

Regarding the MAAPE metric, the same thing does not happen. ARIMA is the algorithm with lower average for intermittent and lumpy category. Makridakis et al. [2018] analyzing a subset of 1045 series of M3 Competition without categorization, using statistical and machine learning algorithms, conclude that the six most accurate methods are statistical. Once again the metrics used were not the same, but in our study most of the time machine learning approach appears as the most accurate.

The accuracy of the model can not be the only concern. As these methods can be applied

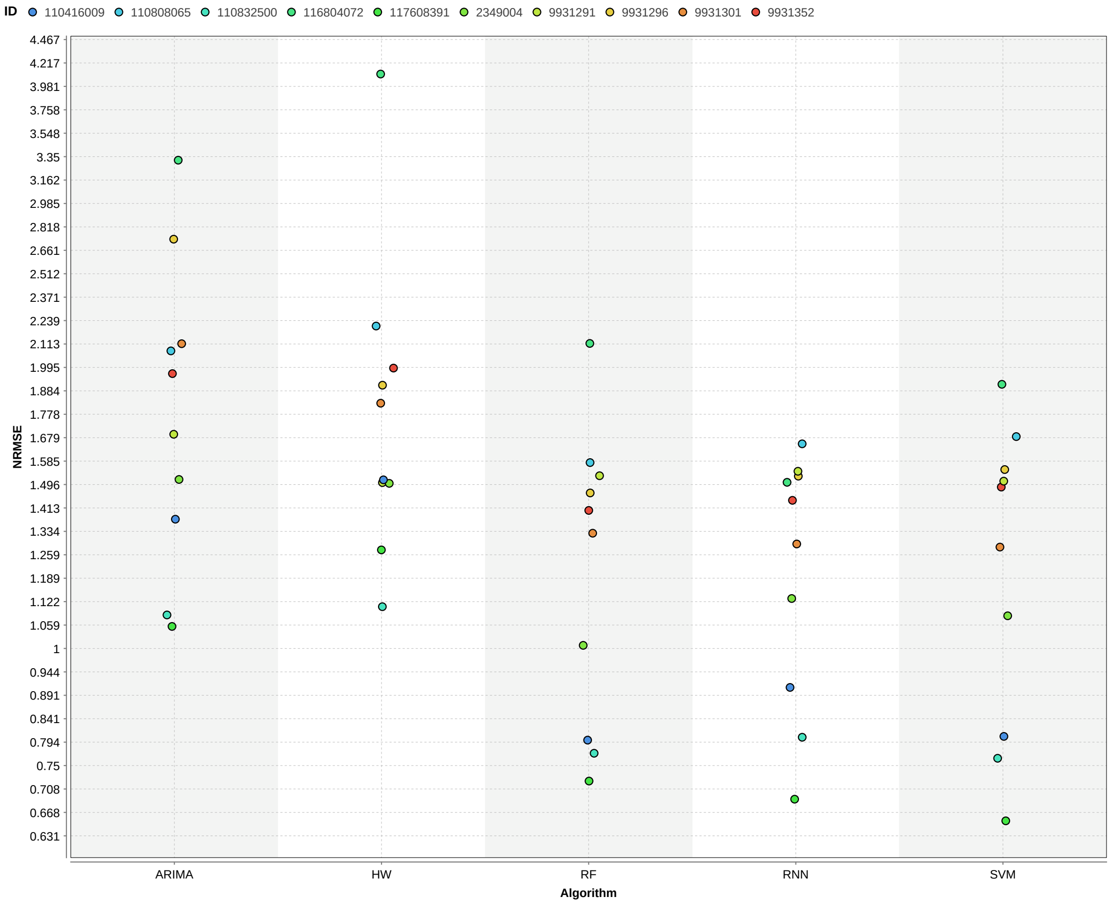


Figure 4.10: Performance of algorithms for Intermittent Category using NRMSE

in many applications with different available resources and restrictions, the computational time becomes critical. It would be be impossible to use RNN to forecast the demand of a data set of thousands of items. When it comes the time to decide the algorithm to use, it will exist a trade-off between accuracy versus computational time. The choice for computational time savings may manifest in a reduction of forecast accuracy.

Chapter 5

Conclusions and Future Work

We have presented, in this dissertation, a comparative study of machine learning and classical statistical methods for demand forecast in a hospital setting. This area deserves more attention. Therefore, a comparative analysis can be a good contribution to integrate the best approaches in decision support systems. The objective was to understand if the application of statistical methods is sufficient to address this problem , or if the introduction of machine learning algorithms can be considered of value. The items have been categorized according to their demand pattern. The goal is to understand if the performance is influenced by the characteristics of the time series.

The results achieved in this dissertation may not be statistically significant, since only ten products for each category have been used, but on the other hand it gives insight into what results can be achieved in a similar and larger study. Regarding RMSE, machine learning algorithms prove to be more accurate than statistical ones. In an inventory management context, using RMSE instead of MAAPE could be more useful because larger errors are undesirable. When MAAPE metric is analyzed, machine learning algorithms only have better results for smooth and erratic categories. By interpreting the aggregated results, we clearly realize that the categorization was useful to conclude that some categories are easier to forecast than others due to their characteristics.

Another important aspect for the algorithm decision is the computational complexity. Machine learning algorithms have larger execution times than statistical methods, which can be a negative factor when choosing an approach, even if this implies a reduction in accuracy.

We can infer that the main goals of the project were accomplished and that the results obtain by the comparison between machine learning and statistical algorithm may be very useful for Knowlogis.

5.1 Future Work

The following suggestions could be followed in order to enrich future research. Firstly is essential to perform a similar study with a larger data set. This is important to have statistical significant results. With few products analyzed, the results could be affected by an sampling errors.

On top of that, study the influence of other variables would be advantageous. Performing different transformations on the original data, in data preparation phase, to understand if machine learning algorithms become more stable and yield better models. Apply transformations to achieve stationary time series, deseasonalizing the data, detrending the data or even a combination of the above are some examples of what could be done. Another interesting case study is to understand if a model has more satisfactory results according to forecast horizon.

In addition, the length of time series may be crucial to yield more accurate models. With larger time series, models could be able to distinguish better noise data and consumption patterns.

References

Luis Aburto and Richard Weber. Improved supply chain management based on hybrid demand forecasts. *Applied Soft Computing*, 7(1):136–144, jan 2007. ISSN 1568-4946. doi: 10.1016/J.ASOC.2005.06.001. URL https://www.sciencedirect.com/science/article/pii/S1568494605000311?via{%}3Dihub.

Jan Adamowski, Hiu Fung Chan, Shiv O. Prasher, Bogdan Ozga-Zielinski, and Anna Sliusarieva. Comparison of multiple linear and nonlinear regression, autoregressive integrated moving average, artificial neural network, and wavelet artificial neural network methods for urban water demand forecasting in Montreal, Canada. *Water Resources Research*, 48(1), jan 2012. ISSN 00431397. doi: 10.1029/2010WR009945. URL http://doi.wiley.com/10.1029/2010WR009945.

Nesreen K. Ahmed, Amir F. Atiya, Neamat El Gayar, and Hisham El-Shishiny. An empirical comparison of machine learning models for time series forecasting. *Econometric Reviews*, 29 (5-6):594–621, 2010. doi: 10.1080/07474938.2010.481556. URL https://doi.org/10.1080/07474938.2010.481556.

JJ Allaire and François Chollet. *keras: R Interface to 'Keras'*, 2018. URL https://CRAN.R-project.org/package=keras. R package version 2.1.5.

JJ Allaire and Yuan Tang. *tensorflow: R Interface to 'TensorFlow'*, 2018. URL https://CRAN.R-project.org/package=tensorflow. R package version 1.5.

Faruk Alpaslan, Erol Eğrioğlu, Çağdaş Hakan Aladağ, and Ebrucan Tiring. An Statistical Research on Feed Forward Neural Networks for Forecasting Time Series. *American Journal of Intelligent Systems*, 2(3):21–25, may 2012. ISSN 2165-8978. doi: 10.5923/j.ajis.20120203.02. URL http://article.sapub.org/10.5923.j.ajis.20120203.02.html.

Atilla Aslanargun, Mammadagha Mammadov, Berna Yazici, and Senay Yolacan. Comparison of arima, neural networks and hybrid models in time series: tourist arrival forecasting. *Journal of Statistical Computation and Simulation*, 77(1):29–53, 2007. doi: 10.1080/10629360600564874.

V. Assimakopoulos and K. Nikolopoulos. The theta model: a decomposition approach to forecasting. *International Journal of Forecasting*, 16(4):521–530, 2000. ISSN 01692070. doi: 10.1016/S0169-2070(00)00066-2. URL http://linkinghub.elsevier.com/retrieve/pii/S0169207000000662.

A T Bon and T K Ng. An Optimization of Inventory Demand Forecasting in University Healthcare Centre. *IOP Conference Series: Materials Science and Engineering*, 166(1):012035, jan 2017. ISSN 1757-8981. doi: 10.1088/1757-899X/166/1/012035.

URL http://stacks.iop.org/1757-899X/166/i=1/a=012035?key=crossref.18fb39f12d79eb68c7c4abf0610616ba.

John Boylan, A A Syntetos, and G C Karakostas. Classification for forecasting and stock control: A case study. *Journal of the Operational Research Society*, 59(4):473–481, 2008. doi: 10.1057/palgrave.jors.2602312. URL https://doi.org/10.1057/palgrave.jors.2602312.

Leo Breiman. Statistical Modeling: The Two Cultures. *Statistical Science*, 16(3):199–231, aug 2001. doi: 10.1214/ss/1009213726. URL http://projecteuclid.org/euclid.ss/1009213726.

Real Carbonneau, Kevin Laframboise, and Rustam Vahidov. Application of machine learning techniques for supply chain demand forecasting. *European Journal of Operational Research*, 184(3):1140–1154, 2008. ISSN 03772217. doi: 10.1016/j.ejor.2006.12.004. URL www.sciencedirect.com/science/article/pii/S0377221706012057?via{%}3Dihub.

Pei-Chann Chang, Yen-Wen Wang, and Chi-Yang Tsai. Evolving neural network for printed circuit board sales forecasting. *Expert Systems with Applications*, 29(1):83–92, jul 2005. ISSN 0957-4174. doi: 10.1016/J.ESWA.2005.01.012. URL https://www.sciencedirect.com/science/article/pii/S0957417405000096.

Chris Chatfield and Mohammed Yar. Prediction intervals for multiplicative Holt-Winters. *International Journal of Forecasting*, 7(1):31–37, may 1991. ISSN 0169-2070. doi: 10.1016/0169-2070(91)90030-Y. URL https://www.sciencedirect.com/science/article/pii/016920709190030Y?via{%}3Dihub.

Chen-Yang Cheng, Kuo-Liang Chiang, and Meng-Yin Chen. Intermittent Demand Forecasting in a Tertiary Pediatric Intensive Care Unit. *Journal of Medical Systems*, 40(10):217, 2016. ISSN 0148-5598. doi: 10.1007/s10916-016-0571-9. URL http://link.springer.com/10.1007/s10916-016-0571-9.

Ching-Wu Chu and Guoqiang Peter Zhang. A comparative study of linear and nonlinear models for aggregate retail sales forecasting. *International Journal of Production Economics*, 86(3):217–231, dec 2003. ISSN 0925-5273. doi: 10.1016/S0925-5273(03)00068-9. URL https://www.sciencedirect.com/science/article/pii/S0925527303000689?via{%}3Dihub{#}BIB25.

Andrey Davydenko and Robert Fildes. Measuring forecasting accuracy: The case of judgmental adjustments to SKU-level demand forecasts. *International Journal of Forecasting*, 29(3):510–522, jul 2013. ISSN 0169-2070. doi: 10.1016/J.IJFORECAST.2012.09.002. URL https://www.sciencedirect.com/science/article/pii/S0169207012001161.

Luis A. Díaz-Robles, Juan C. Ortega, Joshua S. Fu, Gregory D. Reed, Judith C. Chow, John G. Watson, and Juan A. Moncada-Herrera. A hybrid ARIMA and artificial neural networks model to forecast particulate matter in urban areas: The case of Temuco, Chile. *Atmospheric Environment*, 42(35):8331–8340, nov 2008. ISSN 1352-2310. doi: 10.1016/J.ATMOSENV.2008.07.020. URL https://www.sciencedirect.com/science/article/pii/S1352231008006523?via{%}3Dihub.

Max Kuhn. Contributions from Jed Wing, Steve Weston, Andre Williams, Chris Keefer, Allan Engelhardt, Tony Cooper, Zachary Mayer, Brenton Kenkel, the R Core Team, Michael Benesty, Reynald Lescarbeau, Andrew Ziem, Luca Scrucca, Yuan Tang, Can Candan, and Tyler Hunt. *caret: Classification and Regression Training*, 2018. URL https://CRAN.R-project.org/package=caret. R package version 6.0-79.

M Gaur, S Goel, and E Jain. Comparison between nearest Neighbours and Bayesian network for demand forecasting in supply chain management. *2nd International Conference on Computing for Sustainable Global Development, INDIACom 2015*, pages 1433–1436, 2015. URL https://www.scopus.com/inward/record.uri?eid=2-s2.0-84960928415{&}partnerID=40{&}md5=46ce9ff684d815f3a8b982d2081cf272.

C. C. Holt. Forecasting seasonals and trends by exponentially weighted moving averages. Technical report, Carnigie Institute, 1957.

Rob Hyndman, Christoph Bergmeir, Gabriel Caceres, Leanne Chhay, Mitchell O'Hara-Wild, Fotios Petropoulos, Slava Razbash, Earo Wang, and Farah Yasmeen. *forecast: Forecasting functions for time series and linear models*, 2018. URL http://pkg.robjhyndman.com/forecast. R package version 8.3.

Ashu Jain and Avadhnam Madhav Kumar. Hybrid neural network models for hydrologic time series forecasting. *Applied Soft Computing*, 7(2):585–592, mar 2007. ISSN 1568-4946. doi: 10.1016/J.ASOC.2006.03.002. URL https://www.sciencedirect.com/science/article/pii/S1568494606000317?via{%}3Dihub.

Michael J. Kane, Natalie Price, Matthew Scotch, and Peter Rabinowitz. Comparison of arima and random forest time series models for prediction of avian influenza h5n1 outbreaks. *BMC Bioinformatics*, 15(1):276, Aug 2014. ISSN 1471-2105. doi: 10.1186/1471-2105-15-276. URL https://doi.org/10.1186/1471-2105-15-276.

Alexandros Karatzoglou, Alex Smola, Kurt Hornik, and Achim Zeileis. kernlab – an S4 package for kernel methods in R. *Journal of Statistical Software*, 11(9):1–20, 2004. URL http://www.jstatsoft.org/v11/i09/.

Neda Khalil Zadeh, Mohammad Mehdi Sepehri, and Hamid Farvaresh. Intelligent Sales Prediction for Pharmaceutical Distribution Companies: A Data Mining Based Approach. *Mathematical Problems in Engineering*, 2014:1–15, may 2014. ISSN 1024-123X. doi: 10.1155/2014/420310. URL http://www.hindawi.com/journals/mpe/2014/420310/.

Mehdi Khashei and Mehdi Bijari. An artificial neural network (p, d, q) model for time-series forecasting. *Expert Systems with Applications*, 37(1):479–489, jan 2010. ISSN 0957-4174. doi: 10.1016/J.ESWA.2009.05.044. URL https://www.sciencedirect.com/science/article/pii/S0957417409004850?via{%}3Dihub.

Sungil Kim and Heeyoung Kim. A new metric of absolute percentage error for intermittent demand forecasts. *International Journal of Forecasting*, 32(3):669–679, jul 2016. ISSN 0169-2070. doi: 10.1016/J.IJFORECAST.2015.12.003. URL https://www.sciencedirect.com/science/article/pii/S0169207016000121.

Andrew Kusiak, Anoop Verma, and Xiupeng Wei. A data-mining approach to predict influent quality. *Environmental Monitoring and Assessment*, 185(3):2197–2210, Mar 2013. ISSN 1573-2959. doi: 10.1007/s10661-012-2701-2. URL https://doi.org/10.1007/s10661-012-2701-2.

Ik-Whan G. Kwon, Sung-Ho Kim, and David G. Martin. Healthcare supply chain management; strategic areas for quality and financial improvement. *Technological Forecasting and Social Change*, 113:422–428, dec 2016. ISSN 0040-1625. doi: 10.1016/J.TECHFORE.2016.07.014. URL https://www.sciencedirect.com/science/article/pii/S0040162516301585.

Andy Liaw and Matthew Wiener. Classification and regression by randomforest. *R News*, 2(3): 18–22, 2002. URL https://CRAN.R-project.org/doc/Rnews/.

Spyros Makridakis, Evangelos Spiliotis, and Vassilios Assimakopoulos. Statistical and machine learning forecasting methods: Concerns and ways forward. *PLOS ONE*, 13(3):1–26, 03 2018. doi: 10.1371/journal.pone.0194889. URL https://doi.org/10.1371/journal.pone.0194889.

R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2012. URL http://www.R-project.org/. ISBN 3-900051-07-0.

R. Samsudin, A. Shabri, and P. Saad. A Comparison of Time Series Forecasting using Support Vector Machine and Artificial Neural Network Model. *Journal of Applied Sciences*, 10(11): 950–958, nov 2010. ISSN 18125654. doi: 10.3923/jas.2010.950.958. URL http://www.scialert.net/abstract/?doi=jas.2010.950.958.

Renato Cesar Sato. Disease management with ARIMA model in time series. *Einstein (Sao Paulo, Brazil)*, 11(1):128–31, 2013. ISSN 2317-6385. doi: 10.1590/S1679-45082013000100024. URL http://www.ncbi.nlm.nih.gov/pubmed/23579758http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=PMC4872983.

Edward A. Silver, David F. Pyke, and Rein Peterson. *Inventory Management and Production Planning and Scheduling*. John Wiley & Sons, third edition, 1998. ISBN 0471119474.

Zhan-Li Sun, Tsan-Ming Choi, Kin-Fan Au, and Yong Yu. Sales forecasting using extreme learning machine with applications in fashion retailing. *Decision Support Systems*, 46(1):411–419, dec 2008. ISSN 0167-9236. doi: 10.1016/J.DSS.2008.07.009. URL https://www.sciencedirect.com/science/article/pii/S0167923608001371?via{%}3Dihub.

Aris A. Syntetos, Konstantinos Nikolopoulos, and John E. Boylan. Judging the judges through accuracy-implication metrics: The case of inventory forecasting. *International Journal of Forecasting*, 26(1):134–143, 2010. ISSN 01692070. doi: 10.1016/j.ijforecast.2009.05.016. URL http://dx.doi.org/10.1016/j.ijforecast.2009.05.016.

Valerie Tang, Stephen W.Y. Cheng, K. L. Choy, Paul K.Y. Siu, G. T.S. Ho, and H. Y. Lam. An intelligent medical replenishment system for managing the medical resources in the healthcare industry. *2016 IEEE International Conference on Fuzzy Systems, FUZZ-IEEE 2016*, pages 154–161, 2016. doi: 10.1109/FUZZ-IEEE.2016.7737682. URL ieeexplore.ieee.org/document/7737682/.

Li Wang, Haofei Zou, Jia Su, Ling Li, and Sohail Chaudhry. An ARIMA-ANN Hybrid Model for Time Series Forecasting. *Systems Research and Behavioral Science*, 30(3):244–259, may 2013. ISSN 10927026. doi: 10.1002/sres.2179. URL http://doi.wiley.com/10.1002/sres.2179.

Wen-Chuan Wang, Kwok-Wing Chau, Chun-Tian Cheng, and Lin Qiu. A comparison of performance of several artificial intelligence methods for forecasting monthly discharge time series. *Journal of Hydrology*, 374(3-4):294–306, aug 2009. ISSN 0022-1694. doi: 10.1016/J.JHYDROL.2009.06.019. URL https://www.sciencedirect.com/science/article/pii/S0022169409003527{#}bib18.

G.Peter Zhang. Time series forecasting using a hybrid ARIMA and neural network model. *Neurocomputing*, 50:159–175, jan 2003. ISSN 0925-2312. doi: 10.1016/S0925-2312(01)00702-0. URL https://www.sciencedirect.com/science/article/pii/S0925231201007020?via{%}3Dihub.

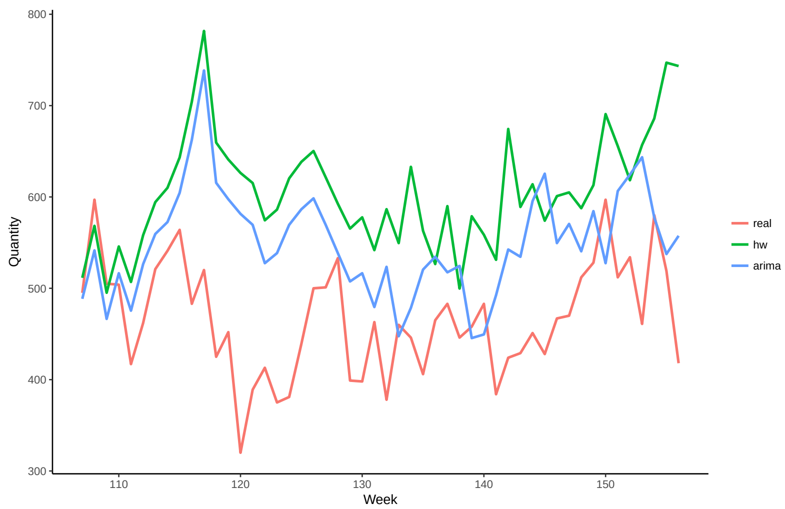
G.Peter Zhang, B.Eddy Patuwo, and Michael Y. Hu. A simulation study of artificial neural networks for nonlinear time-series forecasting. *Computers & Operations Research*, 28(4):381–396, apr 2001. ISSN 0305-0548. doi: 10.1016/S0305-0548(99)00123-9. URL https://www.sciencedirect.com/science/article/pii/S0305054899001239.

Qifeng Zhou, Ruyuan Han, and Tao Li. A two-step dynamic inventory forecasting model for large manufacturing. *Proceedings - 2015 IEEE 14th International Conference on Machine Learning and Applications, ICMLA 2015*, pages 749–753, 2016. doi: 10.1109/ICMLA.2015.93. URL ieeexplore.ieee.org/document/7424411/.

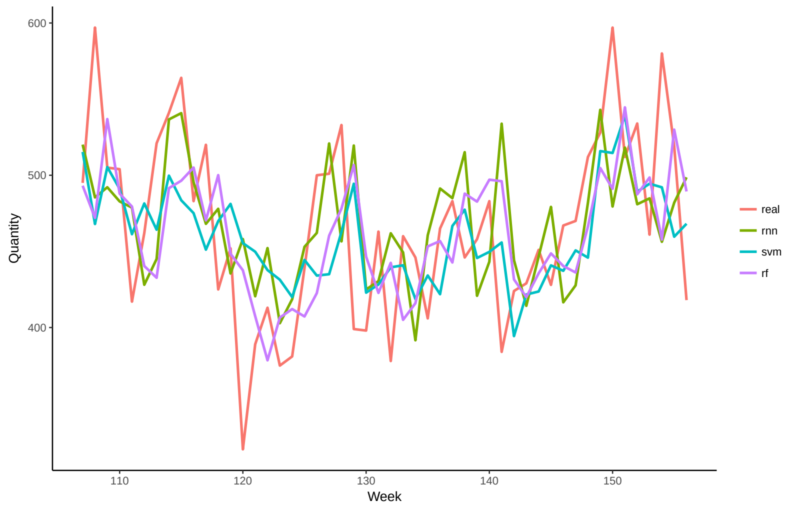
REFERENCES

Appendix A

Forecast for Smooth Category



(a) Statistical Methods



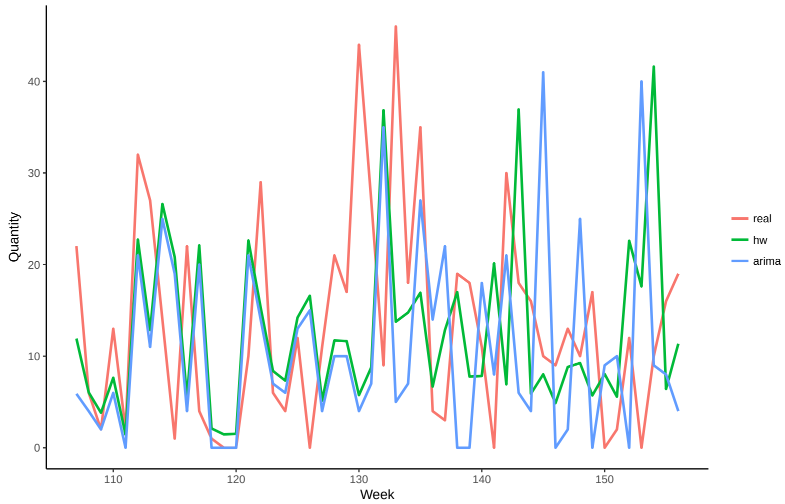
(b) Machine Learning Algorithms

Figure A.1: Forecast of Product no. 110844010

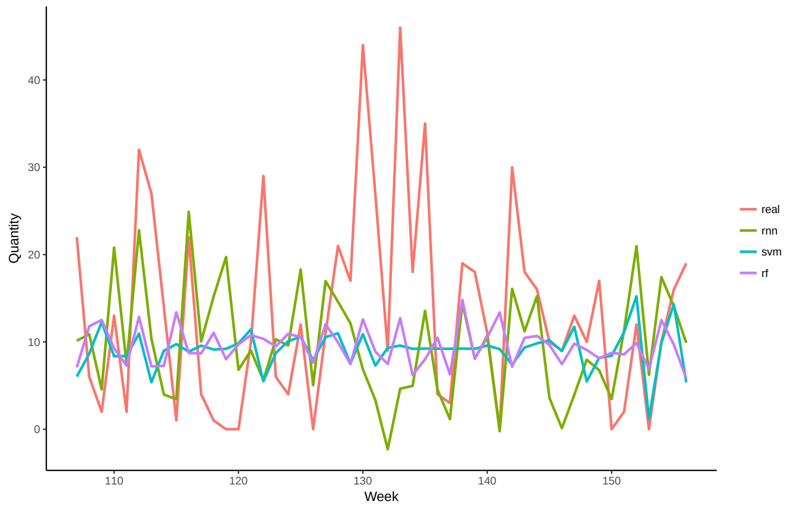
Forecast for Smooth Category

Appendix B

Forecast for Lumpy Category



(a) Statistical Methods



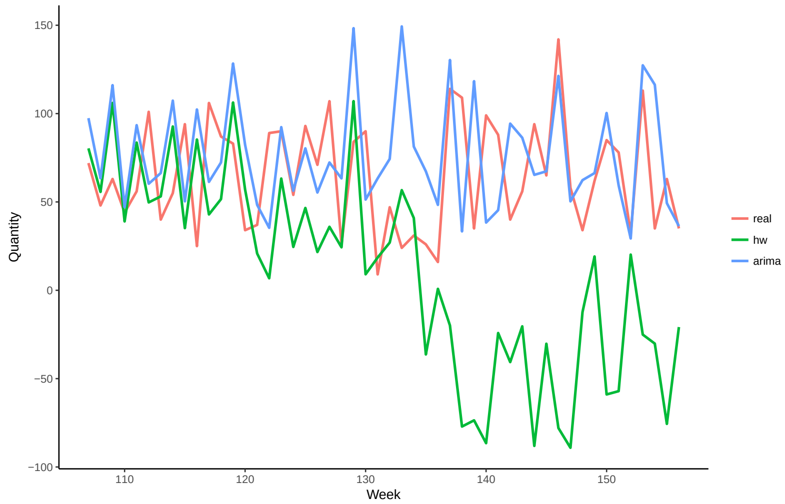
(b) Machine Learning Algorithms

Figure B.1: Forecast of Product no. 110416252

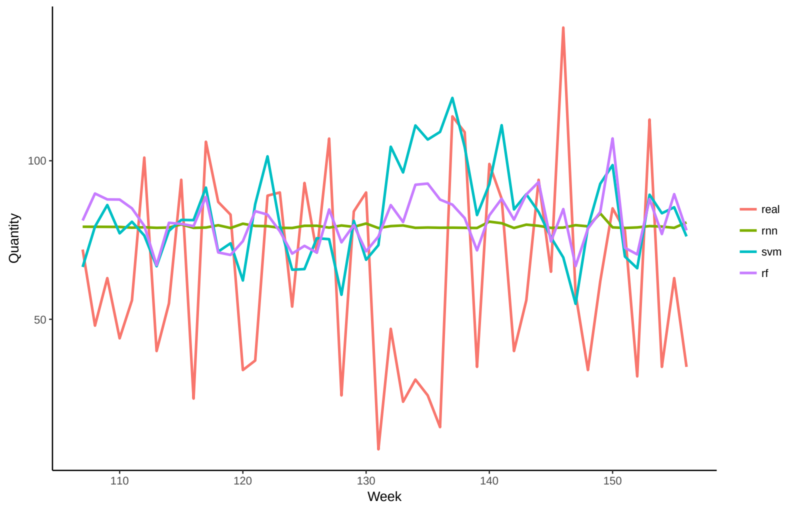
Forecast for Lumpy Category

Appendix C

Forecast for Erratic Category



(a) Statistical Methods



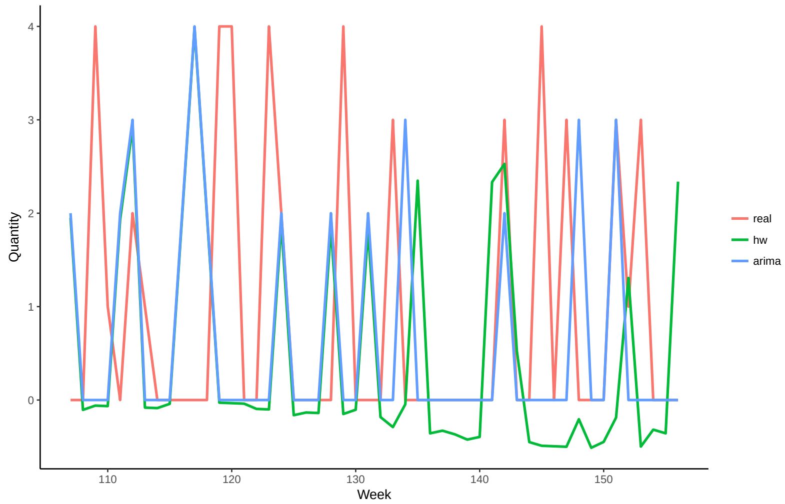
(b) Machine Learning Algorithms

Figure C.1: Forecast of Product no. 110440249

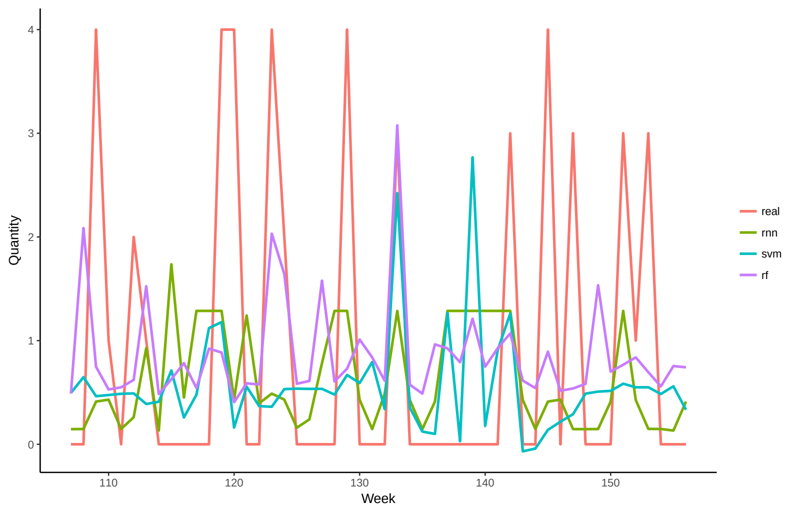
Forecast for Erratic Category

Appendix D

Forecast for Intermittent Category



(a) Statistical Methods



(b) Machine Learning Algorithms

Figure D.1: Forecast of Product no. 110808065